

LNCSE 2091

Josef Bigun
Fabrizio Smeraldi

Audio- and Biometric Person Aut

Third International Confer
Halmstad, Sweden, June 20
Proceedings

Lecture Notes in Computer Science
Edited by G. Goos, J. Hartmanis and J. van Leeuwen

2091

Springer

Berlin

Heidelberg

New York

Barcelona

Hong Kong

London

Milan

Paris

Singapore

Tokyo

Josef Bigun Fabrizio Smeraldi (Eds.)

Audio- and Video-Based Biometric Person Authentication

Third International Conference, AVBPA 2001
Halmstad, Sweden, June 6-8, 2001
Proceedings



Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany
Juris Hartmanis, Cornell University, NY, USA
Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editors

Josef Bigun
Fabrizio Smeraldi
Halmstad University
School of Information Science,
Computer and Electrical Engineering
P.O. Box 823, S-301 18 Halmstad, Sweden
E-mail: {josef.bigun/Fabrizio.smeraldi}@ide.hh.se

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Audio- and video-based biometric person authentication : third
international conference ; proceedings / AVBPA 2001, Halmstad, Sweden,
June 6 - 8, 2001. Josef Bigun ; Fabrizio Smeraldi (ed.). - Berlin ;
Heidelberg ; New York ; Barcelona ; Hong Kong ; London ; Milan ; Paris ;
Singapore ; Tokyo : Springer, 2001
(Lecture notes in computer science ; Vol. 2091)
ISBN 3-540-42216-1

CR Subject Classification (1998): I.5, I.4, I.3, K.6.5, K.4.4, C.2.0

ISSN 0302-9743

ISBN 3-540-42216-1 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

<http://www.springer.de>

© Springer-Verlag Berlin Heidelberg 2001
Printed in Germany

Typesetting: Camera-ready by author

Printed on acid-free paper SPIN: 10839388 06/3142 5 4 3 2 1 0

Preface

This book collects the research work presented at the Third International Conference on Audio- and Video- Based Biometric Person Authentication that took place in Halmstad, Sweden, in June 2001.

As in the preceding two cases, the conference announcement met with a consistent positive response both from industry and the research community. Since 1997, when the first conference took place, the field of Biometric Person Authentication has witnessed the development of commercial solutions that testify the practical relevance of the subject. On the other hand, the high quality of the research papers collected in this volume confirms the scientific importance and the challenging nature of the problems underlying this multi-disciplinary research field.

The volume represents a necessarily concise survey of state-of-the-art techniques in the field and addresses the topics:

- Face as biometrics.
- Face image processing.
- Speech as biometrics and speech processing.
- Fingerprints as biometrics.
- Gait as biometrics.
- Hand, signature, and iris as biometrics.
- Multi-modal analysis and system integration.

Compared to the previous editions, fingerprints and gait have gained emphasis. The book also includes three invited contributions:

- Anil Jain (Michigan State University, USA),
- Josef Kittler (University of Surrey, UK), and
- Satoshi Nakamura (ATR, Japan).

We believe that a sizable contribution of the proceedings resides in its multi-disciplinary character. Growing demands for conjugating security with the mobility and flexibility required by emerging applications, e.g. mobile electronic commerce, can only be addressed through a close cooperation between the communication and the computer science communities. It is likely that multi-modality will play a key role in future authentication systems, which will afford a high degree of robustness to shifting usage conditions and adaptability to scalable security requirements.

We gratefully acknowledge the contributions of the Program Committee, the referees, as well as the sponsoring organizations.

Organization

The international conference AVBPA 2001 was organized by

- the **School of Information Science, Computer and Electrical Engineering**, of Halmstad University, Sweden,
- **TC-14** of IAPR (International Association for Pattern Recognition).

Executive Committee

Conference Chair: Josef Bigun; Halmstad University

Program Chairs: Josef Bigun and Fabrizio Smeraldi; Halmstad University

Local Organization: Eva Nestius, Ulrika Hult, and Ulla Johansson; Halmstad University

Program Committee

The *Program Chairs*,

Frederic Bimbot

Mats Blomberg

Jean-Francois Bonastre

Gunilla Borgefors

Roberto Brunelli

J. M. Hans du Buf

Horst Bunke

Rama Chellapa

Gerard Chollet

Robert Frischholz

Sadaoki Furui

Dolores Garcia-Plaza Cuellar

Dominique Genoud

Bjorn Granstrom

Kenneth Jonsson

Jurgen Luetttin

John Mason

George Matas

Bruce Millar

Jonathon Phillips

Tomaso Poggio

Nalini Ratha

Gael Richard

Massimo Tistarelli

Harry Wechsler

Thomas Vetter

IRISA, France,

Royal Institute of Technology, Sweden,

Uni. d'Avignon e. d. Pays de Vaucluse, France,

Swedish Univ. of Agricultural Sciences, Sweden,

ITC-irst, Italy,

University of Algarve, Portugal,

University of Bern, Switzerland,

University of Maryland, USA,

CNRS, France,

Dialog Communication Systems AG, Germany,

Tokyo Inst. of Technology, Japan,

Ibermática, Spain,

Nuance Communications, USA,

Royal Institute of Technology, Sweden,

Finger Prints AB, Sweden,

ASCOM, Switzerland,

University of Swansea, UK,

CVUT, Czech Republic,

Australian National University, Australia,

DARPA, USA,

MIT, USA,

IBM, USA,

Philips, France,

University of Genova, Italy,

George Mason University, USA,

University of Freiburg, Germany.

Referees

The Program Committee,

Roberto Cesar

University of Sao Paulo, Brazil,

Tony F. Ezzat

MIT, USA,

Cristina Fernandez Grande

Ibermatica, Spain,

Sami Romdhani

University of Freiburg, Germany,

Miguel Schneider-Fontan

Ibermatica, Spain.

Sponsoring Organizations

Halmstad University,

International Association for Pattern Recognition (IAPR),

VISIT program of the Swedish Foundation for Strategic Research,

Swedish Society for Automated Image Analysis, (SSAB).



Table of Contents

Face as Biometrics

Face Identification and Verification via ECOC	1
<i>Kittler J., Ghaderi R., Windeatt T., and Matas J.</i>	
Pose-Independent Face Identification from Video Sequences	14
<i>Lincoln M.C. and Clark A.F.</i>	
Face Recognition Using Independent Gabor Wavelet Features	20
<i>Liu C. and Wechsler H.</i>	
Face Recognition from 2D and 3D Images	26
<i>Wang Y., Chua C.-S., and Ho Y.-K.</i>	
Face Recognition Using Support Vector Machines with the Feature Set Extracted by Genetic Algorithms	32
<i>Lee K., Chung Y., and Byun H.</i>	
Comparative Performance Evaluation of Gray-Scale and Color Information for Face Recognition Tasks	38
<i>Gutta S., Huang J., Liu C., and Wechsler H.</i>	
Evidence on Skill Differences of Women and Men Concerning Face Recognition	44
<i>Bigun J., Choy K.-W., and Olsson H.</i>	
Face Recognition by Auto-associative Radial Basis Function Network	52
<i>Zhang B.L. and Guo Y.</i>	
Face Recognition Using Independent Component Analysis and Support Vector Machines	59
<i>Déniz O., Castrillón M., and Hernández M.</i>	

Face Image Processing

A Comparison of Face/Non-face Classifiers	65
<i>Hjelmås E. and Farup I.</i>	
Using Mixture Covariance Matrices to Improve Face and Facial Expression Recognitions	71
<i>Thomaz C.E., Gillies D.F., and Feitosa R.Q.</i>	
Real-Time Face Detection Using Edge-Orientation Matching	78
<i>Fröba B. and Küblbeck C.</i>	

Directional Properties of Colour Co-occurrence Features for Lip Location
and Segmentation 84
Chindaro C. and Deravi F.

Robust Face Detection Using the Hausdorff Distance 90
Jesorsky O., Kirchberg K.J., and Frischholz R.W.

Multiple Landmark Feature Point Mapping for Robust Face Recognition . 96
Rajapakse M. and Guo Y.

Face Detection on Still Images Using HIT Maps 102
García Mateos G. and Vicente Chicote C.

Lip Recognition Using Morphological Pattern Spectrum 108
Omata M., Hamamoto T., and Hangai S.

A Face Location Algorithm Robust to Complex Lighting Conditions 115
Mariani R.

Automatic Facial Feature Extraction and Facial Expression Recognition .. 121
Dubuisson S., Davoine F., and Cocquerez J.P.

Speech as Biometrics and Speech Processing

Fusion of Audio-Visual Information for Integrated Speech Processing 127
Nakamura S.

Revisiting Carl Bild's Impostor: Would a Speaker Verification System Foil
Him? 144
Sullivan K.P.H. and Pelecanos J.

Speaker Discriminative Weighting Method for VQ-Based Speaker
Identification 150
Kinnunen T. and Fränti P.

Visual Speech: A Physiological or Behavioural Biometric? 157
Brand J.D., Mason J.S.D., and Colomb S.

An HMM-Based Subband Processing Approach to Speaker Identification . 169
Higgins J.E. and Dampier R.I.

Affine-Invariant Visual Features Contain Supplementary Information to
Enhance Speech Recognition 175
Gurbuz S., Patterson E., Tufekci Z., and Gowdy J.N.

Fingerprints as Biometrics

Recent Advances in Fingerprint Verification (*Invited*) 182
Jain A.K., Pankanti S., Prabhakar S., and Ross A.

Fast and Accurate Fingerprint Verification (Extended Abstract) 192
Udupa U.R., Garg G., and Sharma P.K.

An Intrinsic Coordinate System for Fingerprint Matching	198
<i>Bazen A.M. and Gerez S.H.</i>	
A Triple Based Approach for Indexing of Fingerprint Database for Identification	205
<i>Bhanu B. and Tan X.</i>	
Twin Test: On Discriminability of Fingerprints	211
<i>Jain A.K., Prabhakar S., and Pankanti S.</i>	
An Improved Image Enhancement Scheme for Fingerprint Minutiae Extraction in Biometric Identification	217
<i>Simon-Zorita D., Ortega-Garcia J., Cruz-Llanas S., Sanchez-Bote J.L., and Glez-Rodriguez J.</i>	
An Analysis of Minutiae Matching Strength	223
<i>Ratha N.K., Connell J.H., and Bolle R.M.</i>	
Curvature-Based Singular Points Detection	229
<i>Koo W.M. and Kot A.</i>	
Algorithm for Detection and Elminiation of False Minutiae in Fingerprint Images	235
<i>Kim S., Lee D., and Kim J.</i>	
Fingerprint Classification by Combination of Flat and Structural Approaches	241
<i>Marcialis G.L., Roli F., and Frasconi P.</i>	
Using Linear Symmetry Features as a Pre-processing Step for Fingerprint Images	247
<i>Nilsson K. and Bigun J.</i>	
Fingerprint Classification with Combinations of Support Vector Machines	253
<i>Yao Y., Frasconi P., and Pontil M.</i>	
Performance Evaluation of an Automatic Fingerprint Classification Algorithm Adapted to a Vucetich Based Classification System	259
<i>Bartesaghi A., Gómez A., and Fernández A.</i>	
Quality Measures of Fingerprint Images	266
<i>Shen L.L., Kot A., and Koo W.M.</i>	
Gait as Biometrics	
Automatic Gait Recognition by Symmetry Analysis	272
<i>Hayfron-Acquah J.B., Nixon M.S., and Carter J.N.</i>	
Extended Model-Based Automatic Gait Recognition of Walking and Running	278
<i>Yam C.-Y., Nixon M.S., and Carter J.N.</i>	

EigenGait: Motion-Based Recognition of People Using Image
Self-Similarity 284
BenAbdelkader C., Cutler R., Nanda H., and Davis L.

Visual Categorization of Children and Adult Walking Styles 295
Davis, J.W.

A Multi-view Method for Gait Recognition Using Static Body Parameters 301
Johnsson A.Y. and Bobick A.F.

New Area Based Metrics for Gait Recognition 312
Foster J.P., Nixon M.S., and Prugel-Bennett A.

Hand, Signature, and Iris as Biometrics

On-Line Signature Verifier Incorporating Pen Position, Pen Pressure, and
Pen Inclination Trajectories 318
Morita H., Sakamoto T., Ohishi T., Komiya Y., and Matsumoto T.

Iris Recognition with Low Template Size 324
Sanchez-Reillo R. and Sanchez-Avila C.

RBF Neural Networks for Hand-Based Biometric Recognition 330
Sanchez-Reillo R. and Sanchez-Avila C.

Hand Recognition Using Implicit Polynomials and Geometric Features ... 336
Öden C., Erçil A., Yıldız V.T., Kırmızıtaş H., and Büke B.

Multi-modal Analysis and System Integration

Including Biometric Authentication in a Smart Card Operating System ... 342
Sanchez-Reillo R.

Hybrid Biometric Person Authentication Using Face and Voice Features .. 348
Poh N. and Korczak J.

Information Fusion in Biometrics 354
Ross A., Jain A.K., and Qian J.-Z.

PrimeEye: A Real-Time Face Detection and Recognition System Robust
to Illumination Changes 360
Choi J., Lee S., Lee C., and Yi J.

A Fast Anchor Person Searching Scheme in News Sequences 366
Albiol A., Torres L., and Delp E.J.

Author Index 373

Face Identification and Verification via ECOC

J. Kittler, R. Ghaderi, T. Windeatt, and J. Matas

Centre for Vision, Speech and Signal Processing
University of Surrey, Guildford, Surrey GU2 7XH, UK
{J.Kittler, T.Windeatt}@eim.surrey.ac.uk

Abstract. We propose a novel approach to face identification and verification based on the Error Correcting Output Coding (ECOC) classifier design concept. In the training phase the client set is repeatedly divided into two ECOC specified sub-sets (super-classes) to train a set of binary classifiers. The output of the classifiers defines the ECOC feature space, in which it is easier to separate transformed patterns representing clients and impostors. As a matching score in this space we propose the average first order Minkowski distance between the probe and gallery images. The proposed method exhibits superior verification performance on the well known XM2VTS data set as compared with previously reported results.

1 Introduction

Automatic verification and authentication of personal identity based on biometric measurements has become popular in security applications. Existing commercial systems are exploiting a myriad of biometric modalities including voice characteristics, iris scan and finger print. However, as a source of biometric information, the human face plays a particularly important role as facial images (photographs) not only can easily be acquired but also they convey discriminatory features which are routinely used for recognition by humans without the need for specialist training. This opens the possibility for a close human - machine interaction and cooperation. Should the need arise, human operators may readily be called on to endorse machine decisions, as may be desirable, for instance, at border check points, or for access to high security sites. Furthermore, in comparison with other biometrics, face images can be collected in a natural way during the interaction of the subject with the verification system at the point of access. In contrast to other modalities face imaging also allows continuous verification during the client's access to services.

Unfortunately, the performance of automatic systems for face recognition or verification is often poor. Although a considerable progress has been made over recent years, face recognition and verification is still a challenging task. For this reason one of the recent paradigms has been to use multiple modalities to achieve robustness and improved performance. Typically, one would combine voice and face data [2] to achieve better verification rates (lower false rejection and false acceptance rates). However, the merits of the combination of other modalities including face profile, lip dynamics and 3D face information to name but a few have also been investigated. Although the multimodal approach has been demonstrated to achieve significant improvements, there is still the

need to improve the performance of the constituent biometric subsystems to drive the error rates even lower. Some advances recently reported in this context include [9].

As another direction to gain performance improvements, attempts have been made to combine the outputs of several decision making systems. This approach draws on the results in multiple classifier fusion [10]. By combining several opinions one can reduce the error variance of the outputs of the individual experts and achieve better error rates. In [8] it was shown that by combining the scores of several diverse face verification systems the error rate of the best expert could be reduced by more than 42 %. However, such ad hoc designs of multiple expert systems may not necessarily produce the best solutions.

In this paper we propose a novel method for designing multiple expert face verification systems. It is based on the error correcting output codes (ECOC) approach developed for channel coding. The basic idea is to allocate additional bits over and above the bits required to code the source message in order to provide error correcting capability. In the context of pattern classification the idea implies that each class is represented by a more complex code than the conventional code $Z_{ij} = 0 \quad i \neq j$ and $Z_{ij} = 1 \quad i = j$. The implementation of such error resilient code requires more than the usual number of classifiers.

The main difficulty in applying the ECOC classification method to the problem of face verification is that verification is a two class problem and ECOC is suited exclusively to multiclass problems. We overcome this difficulty by proposing a two stage solution to the verification problem. In the first stage we view the verification task as a recognition problem and develop an ECOC design to generate class specific discriminants. In fact we need only the discriminant for the class of the claimed identity. In the second stage we test the hypothesis that the generated discriminant is consistent with the distributions of responses for the particular client.

The proposed scheme leads to an effective design which exhibits the attractive properties of ECOC classifiers but at the same time it is applicable to the two class personal identity verification problem. The design approach has been tested on the XM2VTS face database using the Lausanne protocol. The false rejection and false acceptance rates achieved are superior to the best reported results on this database to date [12].

The paper is organised as follows. In Section 2 we describe how face images are represented. In Section 3 we outline the Error Correcting Output Code method and adapt it to the verification problem. In Section 4 we develop two hypothesis testing approaches which are the basis of the final stage of the verification process. The results of the proposed method obtained on the XM2VTS face database are reported in Section 5 which is followed by conclusions in Section 6.

2 Face Image Representation

Normalisation or standardisation is an important stage in face recognition or verification. Face images differ in both shape and intensity, so *shape alignment* (geometric normalisation) and *intensity correction* (photometric normalisation) can improve performance of the designed system. Our approach to geometric normalisation has been based on eye position. Four parameters are computed from the eye coordinates (rota-

tion, scaling and translation in horizontal and vertical directions) to crop the face part from the original image and scale it to any desired resolution. Here we use “manually localised” eye coordinates to eliminate the dependency of the experiments on processes which may lack robustness. In this way, we can focus our investigation on how the performance is affected by the methodology of verification and in particular by the ECOC technique. For photometric normalisation we have used histogram equalisation as it has exhibited better performance in comparison with other existing methods[12].

Although it is possible to use gray levels directly, as demonstrated in earlier experiments [19, 15], normally features are first extracted. There are many techniques in the pattern recognition literature for extracting and selecting effective features that provide maximal class separation in the feature space [3]. One popular approach is *Linear Discriminant Analysis (LDA)* which is used in our experiments. We briefly review the theory of LDA, and how it is applied to face recognition or verification. Further details may be found in [3] and [14].

Given a set of vectors $x_i \in \mathbb{R}^D$, $x_i \in \mathbb{R}^D$, each belonging to one of c classes $\{C_1, C_2, \dots, C_c\}$, we compute the between-class scatter matrix, S_B ,

$$S_B = \sum_{i=1}^c (\theta_i \otimes \theta)(\theta_i \otimes \theta)^T \quad (1)$$

and within-class scatter matrix, S_W

$$S_W = \sum_{i=1}^c \sum_{x_k \in C_i} (x_k \otimes \theta_i)(x_k \otimes \theta_i)^T \quad (2)$$

where θ is the grand mean and θ_i is the mean of class C_i .

The objective of LDA is to find the transformation matrix, W_{opt} , that maximises the ratio of determinants $\frac{W^T S_B W}{W^T S_W W}$. W_{opt} is known to be the solution of the following eigenvalue problem [3]:

$$S_B W \otimes S_W W \otimes I = 0 \quad (3)$$

Premultiplying both sides by S_W^{-1} , (3) becomes:

$$(S_W^{-1} S_B) W = W \Lambda \quad (4)$$

where Λ is a diagonal matrix whose elements are the eigenvalues of matrix $S_W^{-1} S_B$. The column vectors w_i ($i = 1, \dots, c \otimes 1$) of matrix W are referred to as *fisherfaces* in [11].

In high dimensional problems (e.g. in the case where x_i are images and D is 10^5) S_W is almost always singular, since the number of training samples M is much smaller than D . Therefore, an initial dimensionality reduction must be applied before solving the eigenvalue problem in (4). Commonly, dimensionality reduction is achieved by Principal Component Analysis [21, 10]; the first $(M \otimes c)$ eigenprojections are used to represent vectors x_i . The dimensionality reduction also allows S_W and S_B to be efficiently calculated. The optimal linear feature extractor W_{opt} is then defined as:

$$W_{opt} = W_{lda} \otimes W_{pca} \quad (5)$$

where W_{pca} is the PCA projection matrix and W_{lda} is the optimal projection obtained by maximising

$$W_{lda} = \arg \max_W \frac{W^T W_{pca}^T S_W W_{pca} W}{W^T W_{pca}^T S_B W_{pca} W} \quad (6)$$

3 ECOC Fundamentals

Error-Correcting Output Coding (ECOC) is an information theoretic concept which suggests that there may be advantages in employing ECOC codes to represent different signals which should be distinguished from each other after being corrupted while passing through a transmission channel. Dietterich and Bakiri [4] suggest that classification can be modelled as a transmission channel consisting of “input features”, “training samples”, and “learning paradigm”. Classes are represented by *code words* with large Hamming distance between any pair. ECOC is believed to improve performance both by decomposing the multi-class problem as well as by correcting errors in the decision-making stage [5]. The binary values in the code word matrix are determined by the code generation procedure; it is possible to choose values that provide a meaningful decomposition [20], but usually there is no meaning attached [5, 6, 23, 11]. There are a few methods to find a set of code words with a guaranteed minimum distance between any pair, the most popular being the BCH codes [5, 18], which we use in our experiments.

To understand the ECOC algorithm, consider a $k \times b$ code word matrix Z (k is the number of classes) in which the k rows represent code words (labels), one for each class. In the training phase, for each column, the patterns are re-labelled according to the binary values (“1s” and “0s”), thereby defining two *super classes*. A binary classifier is trained b times, once for each column. Each pattern can now be transformed into ECOC feature space by the b classifiers, giving a vector

$$\underline{y} = [y_1, y_2, \dots, y_b]^T \quad (7)$$

in which y_j is the real-valued output of j th classifier. In the test phase, the distance between output vector and label for each class is determined by

$$L_i = \sum_{j=1}^b Z_{i,j} \otimes y_j \quad (8)$$

and a pattern is assigned to the class corresponding to the code word having minimum distance to \underline{y} .

4 ECOC for Verification

In this section we discuss how the decision making strategy based on ECOC can be modified for the face verification task, which is characterised by a large number of two-class problems with a few training patterns for each client. As explained in Section 1, decision-making in the original ECOC multiple classifier is based on the distance, L_i between the output of its constituent binary classifiers and the code words (compound

labels), which act as representatives of the respective classes. The test pattern is then assigned to the class for which the distance L_i is minimum.

In the case of verification, the task is somewhat different. We wish to ascertain whether the classifier outputs are jointly consistent with the claimed identity. This could be accomplished by setting a threshold on the distance of the outputs from the client code. However, the compound code represents an idealised target, rather than the real distribution of these outputs. Thus measuring the distance from the client code could be misleading, especially in spaces of high dimensionality.

One alternative would be to adopt the *centroid* of the joint classifier outputs to characterise each client and to measure the consistency of a new client claim from this representation. Incidentally, the use of centroid in the context of ECOC classifiers is also advocated in [7]. However, as we have only a very small number of training samples, the estimated centroid would be very unreliable. We propose to represent each client i by a set Y_i of N ECOC classifier output vectors, i.e.

$$Y_i = \{y_i^l\}_{l=1}^N = 1 \times 2^c \times N \times \text{dim} \quad (9)$$

where N is the number of i th client patterns available for training. In order to test the hypothesis that the client claim is authentic we adopt as a test statistic the average distance between vector y and the elements of set Y_i . The distance is measured using first order Minkowski metric, i.e.

$$d_i(y) = \frac{1}{N} \sum_{l=1}^N \sum_{j=1}^b |y_j^l - y_j| \quad (10)$$

where y_j is the j th binary classifier output for the test pattern, and y_j^l is the j th classifier output for the l th member of class i . The distance is checked against a decision threshold, t . If the distance is below the threshold, client's claim is accepted, otherwise it is rejected, i.e.

$$d_i(y) \begin{cases} \leq t & \text{accept claim} \\ > t & \text{reject claim} \end{cases} \quad (11)$$

It should be noted that the measure in (10) can also be used for identification by finding the argument i for which the the distance $d_i(y)$ is minimum, i.e.

$$\text{assign } y \text{ to class } i \text{ if } d_i(y) = \min_j d_j(y) \quad (12)$$

Regardless of whether it is used in the identification or verification mode, we shall refer to the ECOC algorithm deploying measure (10) as multi-seed ECOC.

It is also interesting to note that ECOC can be interpreted as a version of *stacked generaliser* in which level zero multiple classifiers are binary and at level one we have an appropriate classifier for the ultimate task - verification or identification [22]. Although nearest neighbour classifiers advocated for level one by Skalak [22] have exhibited good performance in many applications, they do not perform well when the number of patterns is too low. Our approach is to use the decision rules in (11) and (12) that are based on average distance instead. The motivation for using first order Minkowski metric as in (8) rather than second order (Euclidean metric) is the greater robustness of the former to outliers (highly erroneous outputs of the level zero binary classifiers).

Note that instead of measuring the distance between points, we could measure a between point similarity which can be expressed by a kernel function that assumes a maximum when the distance is zero and monotonically decreases as the distance increases. The design of the decision function cannot involve any training as the number of points available is extremely small. We simply use exponential kernels with fixed width σ . The centres do not need to be explicitly determined because we use $d_i(\underline{y})$ in the exponent of the kernel to measure similarity of \underline{y} to class i . We allocate one kernel per client and a number of kernels for each imposter. We measure the relative similarities of a test vector to the claimed identity and to the impostors as

$$k_i(\underline{y}) = \frac{1}{\sum_{\square} w_{\square} \exp\left(-\frac{d_{\square}(\underline{y})}{\sigma}\right)} \quad (13)$$

where index \square runs over all imposter kernel placements and client i , the weights w_{\square} are estimated and σ^2 defines the width of the kernel. The client claim test is carried out as follows:

$$k_i(\underline{y}) \begin{cases} \geq 0.5 & \text{accept claim} \\ < 0.5 & \text{reject claim} \end{cases} \quad (14)$$

5 Experiments on XM2VTS Data Base

The aim of the experiments reported in this section is to evaluate the proposed approach to personal identity verification and to compare the results with other verification methods. We use the XM2VTS face database for this purpose as it is known to be challenging and several results of experiments, carried out according to an internationally agreed protocol using other verification methods, are readily available in the literature.

5.1 Database and Experimental Protocol

The extended M2VTS (XM2VTS) database contains 295 subjects. The subjects were recorded in four separate sessions uniformly distributed over a period of 5 months, and within each session a number of shots were taken including both frontal-view and rotation sequences. In the frontal-view sequences the subjects read a specific text (providing synchronised image and speech data), and in the rotation sequences the head was moved vertically and horizontally (providing information useful for 3D surface modelling of the head). Further details of this database can be found in [16].¹

The experimental protocol (known as Lausanne evaluation protocol) provides a framework within which the performance of vision-based (and speech-based) person authentication systems running on the extended M2VTS database can be measured. The protocol assigns 200 clients and 95 impostors. Two shots of each session for each subject's frontal or near frontal images are selected to compose two configurations. We used the first configuration which is more difficult as the reported results show [14]. In this configuration, for each client there are 3 training, 3 evaluation and 2 test images. The impostor set is partitioned into 25 evaluation and 70 test impostors. Within the

¹ <http://www.ee.surrey.ac.uk/Research/VSSP/xm2fdb.html>

protocol, the verification performance is measured using the false acceptance and the false rejection rates. The operating point where these two error rates equal each other is typically referred to as the equal error rate point. Details of this protocol can be found in [13].¹

5.2 System Description

All images are projected to a lower dimensional feature space as described in section 2, so that each pattern is represented by a vector with 199 elements. There are 200 clients, so from the identification viewpoint we are facing a 200 class problem. We use a BCH equi-distance code containing 200 codewords (compound labels) 511 bit long. The Hamming distance between any pair of labels is 256 bits. The choice of code and advantages of equi-distance code are discussed in [23].

For the verification task, the level-zero classifier is a Multi-Layer Perceptron (MLP) with one hidden layer containing 199 input nodes, 35 hidden nodes and two output nodes. The Back-propagation algorithm with fixed learning rate, momentum and number of epochs is used for training. The dual output is mapped to a value between “0” and “1” to give an estimation of probability of super-class membership. For the identification task, we used an MLP with three hidden nodes.

As explained in Section 3, the outputs of the MLPs define an ECOC feature vector, and from equation (10), $d_i(\underline{y})$ for the claimed identity i is calculated by averaging over respective class images. In identification, we simply assign to the test vector \underline{y} the class with minimum average distance $d_i(\underline{y})$. For verification we use the two different combining methods described in Section 4, and in both cases we attempt to minimise the error rates on the evaluation set of clients and impostors.

5.3 Identification

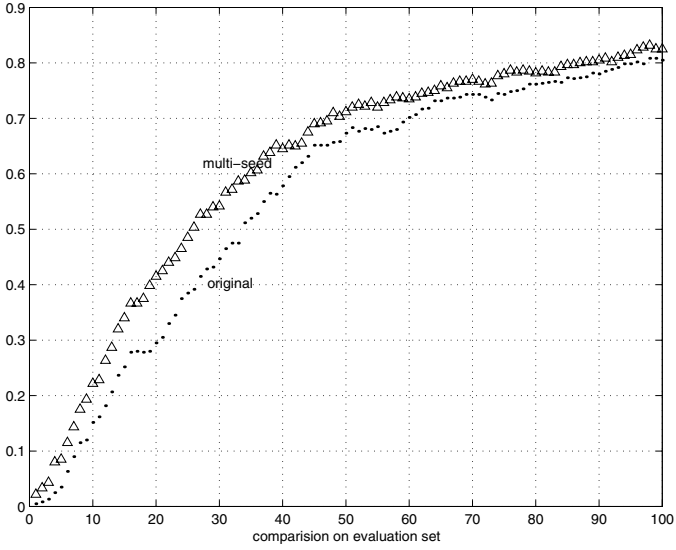
From the identification viewpoint, XM2VTS is a difficult database because the number of classes is high (200), the feature set has a high dimensionality (199 features) and the number of training samples is low (three samples for each subject).

The goal of this experiment is to show that the ECOC technique can solve a complex problem using simple learning machines (a neural network with a few hidden nodes), and to compare the original and the multi-seed ECOC. In identification the evaluation set is not used. We report the results separately for the evaluation and test sets. The rate of correct classification is presented in table II. For shorter codes (0 to 100) in particular, the performance of multi-seed ECOC is better, as clearly shown in figure II.

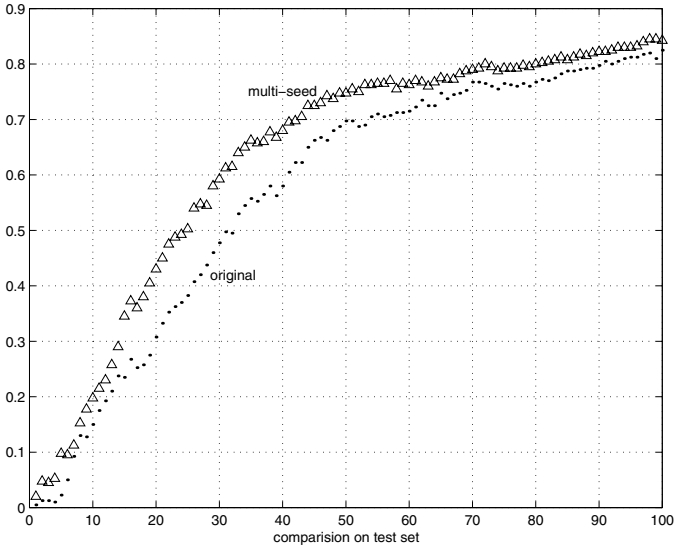
5.4 Verification

Both distance and similarity based rules for combining the outputs of the ECOC multiple classifiers have been investigated. Of the two decision functions, the distance based rule is the only one that depends on a parameter, the decision threshold, that has to be selected.

² http://www.idiap.ch/~m2vts/Experiments/xm2vtsdb_protocol_october.ps



(a)



(b)

Fig. 1. Comparing performance as a function of code length in original and multi-seed ECOC for face identification over a) Evaluation set. b) Test set.

Table 1. Comparison of original and multi-seed ECOC recognition rates (%) obtained in a subject identification experiment using an equi-distance code (200 \times 511).

data set	evaluation	test
Original ECOC	91.83	92.50
Multi-seed ECOC	92.25	93.25

Distance Based Combination. Normally one would use the evaluation set data to compute the Receiver Operating Characteristics (ROC) curve which plots the relationship of false rejection rate and false acceptance rate as a function of threshold. A suitable threshold is then selected to achieve the required behaviour. For instance, one can specify the threshold that delivers equal false rejection and false acceptance rates. The threshold can be selected for each client separately, or globally by averaging the errors over all the clients.

One of the difficulties encountered with our ECOC based approach was that because the level-zero classifier was "too powerful", the FR and FA errors on the evaluation set were zero for a large range of thresholds. In such circumstances the ROC curve is not very useful in threshold setting. This problem was circumvented by the following procedure. Starting from $t = 0$ we successively increased the threshold in fixed steps to find the point where the total error (the sum of FR and FA errors) is minimum. If the total error was zero for several such increments the selected threshold would correspond to the point just before the total error would start rising.

The results obtained with the above threshold selection procedure using the evaluation set data are given in Table 2 as a function of step size. As different step sizes

Table 2. Result of verification when the clients in the evaluation set are used as seeds.

search step	FR(Ev)	FA (Ev)	FR(Ts)	FA(Ts)
.25	0	0	13.2500	0.1078
.2	0	0	10.5000	0.1422
.1	0	0	6.5000	0.2772
.05	0	0	5.2500	0.4130
.01	0	0	4.7500	0.6540
.005	0	0	4.7500	0.7111
.001	0	0	4.5000	0.7391

terminate the threshold selection procedure at different destinations from the impostors in the evaluation set the test set performance varies. In table 3 we report error rates when seeds from both the evaluation and training sets are used to set the thresholds. Even though generalisation has improved, it is not clear from the evaluation set performance how to select the best step size. One possibility is to combine the results from all step sizes, and the final row of table 3 shows the result of majority vote combination.

Table 3. Result of verification when the clients in the evaluation and training sets are used as seeds.

search step	FR(Ev)	FA(Ev)	FR(Ts)	FA(Ts)
.2	0	0.065	6.75	.1676
.1	0	0	4.50	.2174
.05	0	0	3.25	.3668
.01	0	0	1.25	.6495
.005	0	0	1.25	.7038
.001	0	0	1.25	.7482
combining	0	0	1.25	.6603

To demonstrate the effectiveness of ECOC we report in Table 4 the result of applying the exhaustive search method directly to the original 199 dimensional feature vectors. Comparing Tables 3 and 4, the benefits of mapping the input data onto the ECOC output vectors are clearly visible. Note also that in this case the evaluation set error rates are non zero, i.e. the population of clients and impostors are overlapping. In this particular case the ROC curve could have been computed but we did not pursue this particular scheme as it was clearly inferior to the ECOC based approach.

Table 4. Result of verification in the fisher face features space.

search step	FR(Ev)	FA (Ev)	FR(Ts)	FA(Ts)
.25	1.67	0.89	16.75	1.105
.2	0.83	1.07	15.25	1.144
.01	0.167	0.33	8.0	1.180
.005	0.167	0.31	8.0	1.239
.001	0.167	0.2925	8.0	1.310

Kernel Combination. Although the kernel combination method requires no thresholds, there are design parameters that can be varied to control the behaviour of the method. In particular, we can choose different ways to represent impostors. Each of the 25 evaluation impostors has 4 sets of 2 images as explained in Section 5.1. Therefore, as an alternative to 25 centres averaged over 4 sets we can choose 50 centres averaged over 2 sets or 100 centres averaged over 1 set. The error rates for 25, 50, 100 impostor centres, along with the results of combining by majority vote are shown in Table 5. In comparison with Table 3, there is a different trade-off between false acceptance and false rejection rates.

Table 5. Result of verification using the kernel score with different numbers of centres for the impostors.

impostor centres	FR(Ev)	FA(Ev)	FR(Ts)	FA(Ts)
25	0	0	0.7500	0.8833
50	0	0	0.5000	0.8786
100	0	0	0.7500	1.2455
combining	0	0	0.7500	0.8596

5.5 Comparison with Other Methods

For comparison we are including the results obtained using three other methods on the same data set and with the same protocol. The methods use the same representation of image data in terms of 199 fisher face coefficients. They employ three different scores for decision making in this feature space. In particular, we use the Euclidean metric, s_E , Normalised correlation, s_N , and Gradient metric, s_O , as detailed in [9]. The results are summarised in Table 6

Table 6. Performance of the three baseline matching scores on manually registered images.

Score	Evaluation set			Test set		
	FR	FA	TE	FR	FA	TE
s_E	7.83	7.83	15.66	5.50	7.35	12.85
s_N	2.50	2.50	5.00	2.25	2.56	4.81
s_O	1.74	1.74	3.48	1.75	1.70	3.45

The results show a number of interesting features. First of all, by comparing the Euclidean metric performance with the proposed distance $d_i(y)$ in Table 4 it would appear that the more robust metric used in $d_i(y)$ combined with the multi-seed representation of clients may be more effective than the Euclidean distance based score. Most importantly, all the ECOC based results are decisively superior to the decision making in the original Fisher face space. Finally, the combination of ECOC multiple classifier outputs by means of the relative similarity score in (14) appears to yield slightly better results than using the distance based score $d_i(y)$. The implication of this finding and of the work reported elsewhere is that the choice of decision (score) function plays an extremely important role in the design of verification systems and should receive more attention in the future.

6 Conclusion

We described a novel approach to face identification and verification based on the Error Correcting Output Coding (ECOC) classifier design concept. In the training phase the client set is repeatedly divided into two ECOC specified sub-sets (super-classes) to train a set of binary classifiers. The output of the classifiers defines the ECOC feature space, in which it is easier to separate transformed patterns representing clients and impostors. As a matching score in the ECOC feature space a novel distance measure and a kernel based similarity measure have been developed. The distance based score computes the average first order Minkowski distance between the probe and gallery images which is more effective than the Euclidean metric. The proposed method was shown to exhibit superior verification performance on the well known XM2VTS data set as compared with previously reported results.

Acknowledgements

The support received from OmniPerception Ltd, EPSRC Grant GR/M61320 and EU Framework V Project Banca is gratefully acknowledged.

References

1. P N Belhumeur, J P Hespanha, and D J Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. In *Proc. of ECCV'96*, pages 45–58, Cambridge, United Kingdom, 1996.
2. S Ben-Yacoub, J Luetttin, K Jonsson, J Matas, and J Kittler. Audio-visual person verification. In *Computer Vision and Pattern Recognition*, pages 580–585, Los Alamitos, California, June 1999. IEEE Computer Society.
3. P.A. Devijver and J. Kittler. *Pattern Recognition: A Statistical Approach*. Prointice Hall, 1982.
4. T.G Dietterich and G. Bakiri. Error-correcting output codes: A general method for improving multiclass inductive learning programs. pages 572–577. Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91), AAAI Pres, 1991.
5. T.G. Dietterich and G Bakiri. Solving multi-class learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995.
6. R. Ghaderi and T. Windeatt. Circular ecoc, a theoretical and experimental analysis. pages 203–206, Barcelona, Spain, September 2000. International Conference of Pattern Recognition (ICPR2000).
7. G. James. *Majority Vote Classifiers: Theory and Applications*. PhD thesis, Dept. of Statistics, Univ. of Stanford, May 1998. <http://www-stat.stanford.edu/~gareth/>.
8. J Kittler, Y P Li, and J Matas. Face verification using client specific fisher faces. In J T Kent and R G Aykroyd, editors, *The Statistics of Directions, Shapes and Images*, pages 63–66, 2000.
9. J Kittler, Y P Li, and J Matas. On matching scores for lda-based face verification. In M Mirmehdi and B Thomas, editors, *British Machine Vision Conference*, 2000.
10. J Kittler and F Roli. *Multiple Classifier Systems*. Springer-Verlag, Berlin, 2000.
11. E.B. Kong and T.G. Diettrich. Probability estimation via error-correcting output coding. Banff, Canada, 1997. Int. Conf. of Artificial Intelligence and soft computing. <http://www.cs.orst.edu/~tgd/cv/pubs.html>.

12. Y.P. Li. *Linear Discriminant Analysis and its application to face Identification*. PhD thesis, School of Electronic Engineering, Information technology and Mathematics, University of Surrey, Guildford, Surrey, U.K. GU2 7X, September 2000.
13. J Luetttin and G. Maitre. *Evaluation Protocol For The Extended M2VTS Database (XM2VTS)*. Dalle Molle Institute for Perceptual Artificial Intelligence, P.O. Box 592 Martigny, Valais, Switzerland, July 1998. IDIAP-Com 98-05.
14. J. Matas, M. Hamouz, M. Jonsson, J. Kittler, Y. Li, C. Kotroupolous, A. Tefas, I. Pitas, T. Tan, H. Yan, F. Smeraldi, J. Bigun, N. Capdevielle, w. Gerstner, S. Ben-Yacoub, Y. Abduljaoued, and Y. Majoraz. Comparison of face verification results on the xm2vts database. In A. A Sanfeliu, J.J Villanueva, M. Vanrell, R. Alqueraz, J. Crowley, and Y. Shirai, editors, *Proceedings of the 15th ICPR*, volume 4, pages 858–863, Los Alamitos, USA, September 2000. IEEE Computer Soc Press.
15. J Matas, K Jonsson, and J Kittler. Fast face localisation and verification. *IVC*, 17(8):578–581, June 1999.
16. K. Messer, J. Matas, J. Kittler, J. Luetttin, and G. Maitre. XM2VTSDB: The extended M2VTS database. In *Proc. of AVBPA'99*, pages 72–77, 1999.
17. M. Nadler and E.P Smith. *Pattern Recognition Engineering*. John Weley and Sons INC., 1993.
18. W.W. Peterson and JR. Weldon. *Error-Correcting Codes*. MIT press, Cambridge,MA, 1972.
19. F.S. Samaria and A.C. Harter. Parameterisation of a stochastic model for human face identification. In *Proceeding of the 2nd IEEE Workshop on application of computer vision*, Sarasota,Florida, 1994. <http://mambo.ucsc.edu/psl/olivetti.html>
20. T.J. Senjnowski and C.R. Rosenberg. Parallel networks that learn to pronounce english text. *Complex systems*, 1(1):145–168, 1987.
21. L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization on human face. *Journal .Opt.Soc. Am. A*, 3(4):519–524, 1987.
22. D.B. Skalak. *Prototype Selection for Composite Nearest Neighbor Classifiers*. PhD thesis, Dept. of Computer Science, Univ. of Massachusetts Amherst, May 1997.
23. T. Windeatt and R. Ghaderi. Binary codes for multi-class decision combining. volume 4051, pages 23–34, Florida,USA, April 2000. 14th Annual International Conference of Society of Photo-Optical Instrumentation Engineers (SPIE).

Pose-Independent Face Identification from Video Sequences

Michael C. Lincoln and Adrian F. Clark

VASE Laboratory, University of Essex, Colchester CO4 3SQ, UK
{mclinc, alien}@essex.ac.uk

Abstract. A scheme for pose-independent face recognition is presented. An “unwrapped” texture map is constructed from a video sequence using a texture-from-motion approach, which is shown to be quite accurate. Recognition of single frames against calculated unwrapped textures is carried out using principal component analysis. The system is typically better than 90% correct in its identifications.

1 Introduction

Face recognition is currently a particularly active area of computer vision. Although work on face analysis was performed as long ago as the 1970s [1], current interest was arguably inspired by the “eigenfaces” technique [2]. Subsequent workers have applied a wide variety of approaches, including various types of neural networks [3], hidden Markov models [4] and shape analysis [5].

The vast majority of face recognition techniques, including all those listed above, concentrate on full-face imagery. This is partly because such a constraint simplifies the problem and partly because typical current applications are for situations in which the subject is cooperative. There has been work on face recognition from profile imagery [6] but the more general problem in which the head orientation is unknown remains relatively unexplored. Full 3D face recognition is touched on in [7] and considered in more detail in [8]. The area in which face recognition technology arguably has the most potential is in policing, where full-face imagery is rarely available. Hence, *pose-independent* schemes are of practical value; indeed, the approach outlined herein is being developed in association with Essex Police.

To be able to perform pose-independent face recognition, one ideally would have images of subjects captured at all possible orientations. This is not a tractable solution; but it is easy to consider an “image” that is a projection of the head shape onto a notional *cylinder* rather than onto a plane. We term this an *unwrapped texture map*. Our scheme involves taking each image (planar projection) in a video sequence, tracking the head from frame to frame and determining the head orientation in each frame, then merging the appropriate region of the image into the unwrapped texture map. If the head exhibits a reasonable amount of motion, a fairly complete texture map can be accumulated. There are similarities between the texture tracker described herein and that reported

in [9], though the two were developed independently. Moreover, [9] did not attempt identification.

Identification schemes applied to conventional, planar images can be exploited on unwrapped texture maps, though care is needed. For example, Kanade-like distances between interior features can be used [1], as can eigen-based approaches, as used here. Most importantly however, one can compare a single frame of a person's head with a portion of a texture map to achieve identification.

This remainder of this paper is organized as follows. The construction of an unwrapped texture map, the most important component of the scheme, is described in Sec 2. The use of these textures in an eigenfaces-like identification scheme is discussed in Sec 3. Conclusions are drawn in Sec 4.

2 Construction of an Unwrapped Texture Map

2.1 Preliminaries

A 3D surface model of the head being tracked is required in order to evaluate the corresponding texture. An accurate model of the head is not required, though poor models are likely to affect the accuracy and stability of tracking. This work employs a tapered ellipsoid as a user-independent head model; this is a simple shape to control and, as it is convex, means that hidden surface removal can be accomplished with back-face culling [10].

In computer graphics, a 2D texture is normally applied to a 3D model. Associated with each vertex in each facet of a 3D model is a 2D texture coordinate. The rendering process then determines the appearance of each screen pixel for each facet by interpolating between the texture coordinates of the vertices. However, this technique requires the reverse operation: values are inserted *into* the texture map when the image positions of the projections of vertices of the head model have been determined. Our implementation of this uses OpenGL, which allows this process to be carried out in hardware, even on PC-class systems.

As explained above, not every pixel in the texture map will have the same accuracy. Hence, each pixel in the constructed texture map has a corresponding confidence value (forming a *confidence map*). This is modeled as the ratio between the area of a pixel in texture space and the area in screen space that gave rise to it. These confidence values are central to the way in which image data are merged into the unwrapped texture map.

Finally, a measure of the similarity between two textures is required. The measure used herein is

$$\sqrt{\frac{\sum_{uv} C_{min}^2(u, v) d^2(u, v)}{\sum_{uv} C_{min}^2(u, v)}} \quad (1)$$

where $d(u, v)$ is the sum-squared difference between textures for the pixel at (u, v) and $C_{min}(u, v)$ is the minimum confidence value for the same pixel.

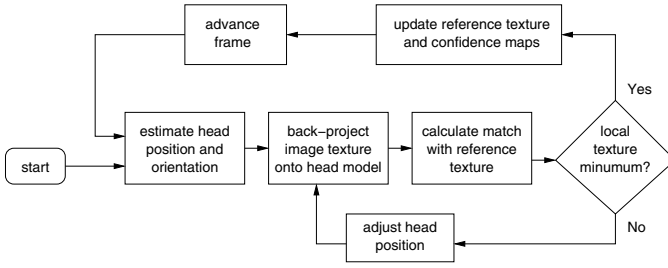


Figure 1. Procedure for constructing unwrapped texture map.

2.2 Tracking by Optimization

The position and orientation of the head in the first frame of a sequence is currently specified manually, though this could be automated. As outlined above, the “reference” unwrapped texture and confidence maps are initialized from the image. The procedure for accumulating the texture and confidence maps from subsequent frames is illustrated in Fig 1. An estimate for the head’s new position and orientation is made; this can be simply the same as in the previous frame, though some prediction scheme (*e.g.*, Kalman filtering) is probably better. The head model is transformed to this new position and the image texture *back-projected* onto it, facet by facet. A match with the reference head texture is then performed. The six position and orientation parameters of the head model are adjusted using a simplex optimization scheme until the best (smallest) match value is obtained. (Simplex seems to be as effective as the more sophisticated scheme described in [9].

With the optimum parameters found, the back-projected texture for the current frame is merged into the reference texture map. A pixel in the texture map is updated only if it will result in a higher confidence value: if C_r is the confidence of a pixel in the reference image and C_i the corresponding value for the current image, then

$$W_r = \frac{C_r}{C_r + C_i} \quad W_i = \frac{C_i}{C_r + C_i} \quad (2)$$

and, providing $C_i > C_r$, the texture map value V_r is updated to

$$V_r = V_r W_r + V_i W_i \quad (3)$$

where V_i is the value of texture map for the current image.

This procedure is illustrated in Fig 2, which shows a single frame of a video sequence. Superimposed at the top right of the frame are the unwrapped texture and confidence maps extracted from that frame, and superimposed at the top left are the corresponding maps accumulated over the entire sequence to date. Note that, for performance reasons, the accumulated texture is constructed

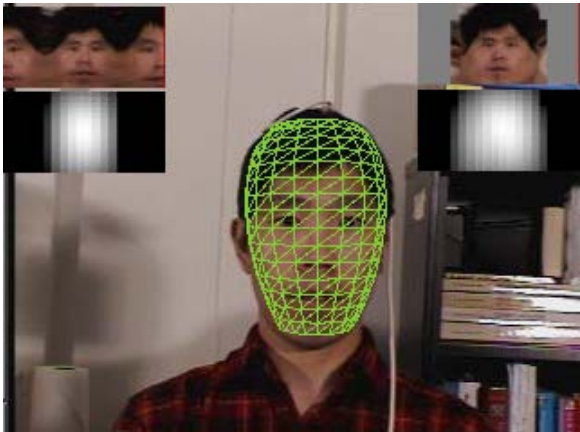


Figure 2. The construction of texture maps by tracking.

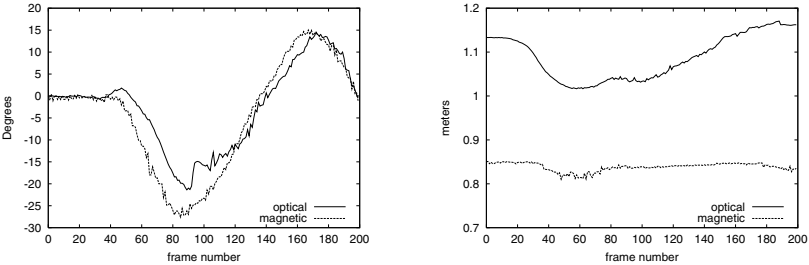


Figure 3. Head-tracking accuracy.

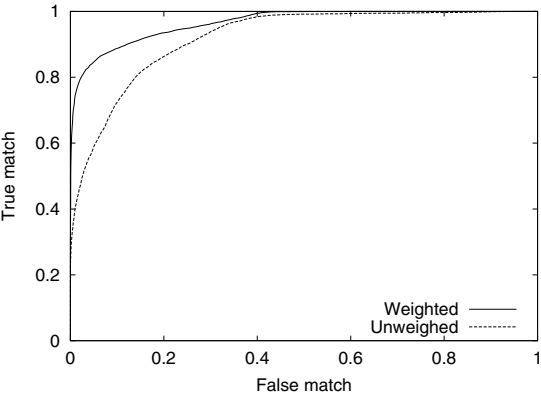


Figure 4. Receiver Operating Characteristic.

without back-face culling and hence appears twice at slightly differing magnifications. The confidence map, however, does employ back-face culling. Fig 3 illustrates the accuracy of the tracker on the “Boston” dataset used in [9], who captured position and orientation information of subjects’ heads as they moved using a magnetic tracker. Compared to the magnetic tracker data, the RMS positional and orientation errors from this texture tracker are 3.5 cm and 2.8° respectively.

3 Recognition Using Derived Texture Maps

To explore recognition, the authors used the “Boston” dataset, which consists of nine 200-frame video sequences for each of five subjects. This is admittedly a small dataset, so the results presented here should be treated with some caution. (A somewhat larger dataset is in the process of being collected by the authors.)

Two forms of the well-established “eigenfaces” recognition scheme [2] have been examined. The first form simply uses contrast normalization before principal component analysis (PCA) but assumes each texture pixel has the same accuracy, which is not the case. The second form weights the variance of each element of each face vector by its corresponding confidence value. Using the notation of [2]:

$$\Psi = \left(\frac{1}{1 + \sum_{j=1}^M \omega_j} \right) \sum_{n=1}^M \Gamma_n \omega_n \quad (4)$$

$$\Phi_i = (\Gamma_i - \Psi_i) \omega_i \quad (5)$$

where ω is the confidence vector for an image.

The test procedure adopted is in accordance with [11], which describes the distinction between “verification” and “identification” in Table 1. Each video sequence was treated as a separate sample, and used only for training, testing, or imposter samples. A “leave-one-out” testing methodology was used, resulting in about 1,800 separate tests. The result of the testing is shown in Fig 4 for both conventional and weighted PCA. Overall performance is summarized in Table 1, which is commensurate with front-face-only techniques. Indeed, the error rate is similar to that of [2], so the extension of the technique to accommodate unwrapped texture maps is introducing no new significant sources of error. It is also apparent that recognition performance is appreciably better using the weighted PCA, justifying the weighting terms introduced in the calculation of the covariance matrix.

4 Conclusions and Further Work

This paper presents initial results from a “texture-from-motion” scheme that is being developed for pose-independent face recognition. The approach to

Table 1. Recognition performance.

Database	Equal error rate (verification)	Error rate (identification)
Boston (tracked, PCA)	16%	6%
Boston (tracked, weighted PCA)	10%	2%

building texture maps appears to be reasonably effective, as demonstrated here. Recognition performance is promising, though not yet comparable to the best front-face-only schemes. However, face-feature normalization has not yet been included in this scheme. It is anticipated that it will improve recognition rates in this scheme, just as it does with conventional, front-face schemes. In the longer term, it is intended to extract shape information at the same time as texture information, and our expectation is that this will lead to a further improvement in performance.

References

1. T. Sakai, M. Nagao, and T. Kanade. Computer analysis and classification of photographs of human faces. In *Proc. First USA-Japan Computer Conference*, pages 55–62, 1972.
2. M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
3. S. Lawrence, C. Giles, A. Tsoi, and A. Back. Face recognition: A convolutional neural network approach. *IEEE Trans. Neural Networks*, 8:98–113, 1997.
4. F. Samaria. Face segmentation for identification using hidden markov models. In *Proceedings of the 1993 British Machine Vision Conference*, pages 399–408, University of Surrey, 1993.
5. A. Lanitis, T. J. Cootes, and C. J. Taylor. An automatic face identification system using flexible appearance models. In *Proceedings of the 1994 British Machine Vision Conference*, pages 65–74, University of York, 1994.
6. A. Samal and P. A. Iyengar. Automatic recognition and analysis of human faces and facial expressions: A survey. *Pattern Recognition*, 25(1):65–77, January 1992.
7. R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, October 1993.
8. S. McKenna, S. Gong, and J. J. Collins. Face tracking and pose representation. In *Proceedings of the 1996 British Machine Vision Conference*, pages 755–764, University of Edinburgh, 1996.
9. Marco La Cascia and Stan Sclaroff. Fast, reliable head tracking under varying illumination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 1999.
10. J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley Systems Programming Series. Addison-Wesley, 1990.
11. Association for Biometrics. Best practices in testing and reporting performance of biometric devices, January 2000.
<http://www.afb.org.uk/bwg/bestprac10.pdf>.

Face Recognition Using Independent Gabor Wavelet Features

Chengjun Liu¹ and Harry Wechsler²

¹ Dept. of Math and Computer Science, University of Missouri, St. Louis, MO 63121
cliu@cs.ums1.edu, <http://www.cs.ums1.edu/~cliu/>

² Dept. of Computer Science, George Mason University, Fairfax, VA 22030
wechsler@cs.gmu.edu, <http://cs.gmu.edu/~wechsler/personal/>

Abstract. We introduce in this paper a novel Independent Gabor wavelet Features (IGF) method for face recognition. The IGF method derives first an augmented Gabor feature vector based upon the Gabor wavelet transformation of face images and using different orientation and scale local features. Independent Component Analysis (ICA) operates then on the Gabor feature vector subject to sensitivity analysis for the ICA transformation. Finally, the IGF method applies the Probabilistic Reasoning Model for classification by exploiting the independence properties between the feature components derived by the ICA. The feasibility of the new IGF method has been successfully tested on face recognition using 600 FERET frontal face images corresponding to 200 subjects whose facial expressions and lighting conditions may vary.

1 Introduction

Face recognition has wide applications in security (biometrics and forensics), human-computer intelligent interaction, digital libraries and the web, and robotics [4], [18]. It usually employs various statistical techniques, such as PCA (principal component analysis) [20], [15], FLD (Fisher linear discriminant, a.k.a. LDA, linear discriminant analysis) [19], [2], [8], ICA (independent component analysis) [1], [7], [14], and Gabor and bunch graphs [21] to derive appearance-based models for classification.

Independent Component Analysis (ICA) has emerged recently as one powerful solution to the problem of blind source separation [5] while its possible use for face recognition has been shown in [1], [7] by using a neural network approximation. ICA searches for a linear transformation to express a set of random variables as linear combinations of statistically independent source variables [5]. The search criterion involves the minimization of the mutual information expressed as a function of high order cumulants. While PCA considers the 2nd order moments only and it uncorrelates the data, ICA would further reduce statistical dependencies and produce an independent code useful for subsequent pattern discrimination and associative recall [16]. ICA thus provides a more powerful data representation than PCA.

The Gabor wavelets, which capture the properties of spatial localization, orientation selectivity, spatial frequency selectivity, and quadrature phase relationship, seem to be a good approximation to the filter response profiles encountered experimentally in cortical neurons [6], [9]. The Gabor wavelets have been found to be particularly suitable

for image decomposition and representation when the goal is the derivation of local and discriminating features. Most recently, Donato et al [7] have experimentally shown that the Gabor filter representation is optimal for classifying facial actions.

This paper introduces a novel Independent Gabor wavelet Features (IGF) method for face recognition. The Gabor transformed face images exhibit strong characteristics of spatial locality, scale and orientation selectivity, similar to those displayed by the Gabor wavelets. Such characteristics produce salient local features, such as the eyes, nose and mouth, that are most suitable for face recognition. The feasibility of the new IGF method has been successfully tested on face recognition using 600 FERET frontal face images corresponding to 200 subjects whose facial expressions and lighting conditions may vary. The effectiveness of the IGF method is shown in terms of both absolute performance indices and comparative performance against some popular face recognition schemes such as the traditional Eigenfaces method and Gabor wavelet based classification method.

2 Gabor Feature Analysis

Gabor wavelets are used for image analysis because of their biological relevance and computational properties [6], [9]. The Gabor wavelets, whose kernels are similar to the 2D receptive field profiles of the mammalian cortical simple cells, exhibit strong characteristics of spatial locality and orientation selectivity, and are optimally localized in the space and frequency domains.

The Gabor wavelets (kernels, filters) can be defined as follows [11], [7]:

$$g_{\theta, \sigma}(z) = \frac{g_{\theta, \sigma}^2}{\sigma^2} e^{\frac{j k_{\theta, \sigma}^2 z^2}{2\sigma^2}} e^{j k_{\theta, \sigma} z} \otimes e^{\frac{j^2}{2}} \quad (1)$$

where θ and σ define the orientation and scale of the Gabor kernels, $z = (x^c y)$, σ denotes the norm operator, and the wave vector $k_{\theta, \sigma}$ is defined as follows:

$$k_{\theta, \sigma} = k_{\sigma} e^{j\theta} \quad (2)$$

where $k_{\sigma} = k_{max}/f^{\sigma}$ and $\sigma = \theta/8$. f is the spacing factor between kernels in the frequency domain [11]. In most cases one would use Gabor wavelets at five different scales, $\sigma \in [0, \dots, 4]$ and eight orientations, $\theta \in [0, \dots, 7]$ [9], [10], [3].

The Gabor wavelet transformation of an image is the convolution of the image with a family of Gabor kernels as defined by Eq. 1. Let $O_{\theta, \sigma}^{(\sigma)}$ denote a normalized convolution output (downsampled by σ and normalized to zero mean and unit variance), then the augmented feature vector $\mathcal{X}^{(\sigma)}$ is defined as follows:

$$\mathcal{X}^{(\sigma)} = \begin{bmatrix} O_{0,0}^{(\sigma)t} & O_{0,1}^{(\sigma)t} & \dots & O_{4,7}^{(\sigma)t} \end{bmatrix}^t \quad (3)$$

where t is the transpose operator. The augmented feature vector thus encompasses all the outputs, $O_{\theta, \sigma}(z)$ ($\theta \in [0, \dots, 7]$ and $\sigma \in [0, \dots, 4]$), as important discriminating information.

3 Independent Component Analysis of the Gabor Features for Face Recognition

We now describe our novel Independent Gabor wavelet Features (IGF) method for face recognition. The augmented Gabor feature vector introduced in Sect. 2 resides in a space of very high dimensionality, and we first apply PCA for dimensionality reduction:

$$\mathcal{Y}^{(\square)} = P^t \mathcal{X}^{(\square)} \quad (4)$$

where $P = [P_1 P_2 \lll P_n]$ consists of the n eigenvectors corresponding to the leading eigenvalues of the covariance matrix of $\mathcal{X}^{(\square)}$, $n < N$ and $P \in \mathbb{R}^{N \times n}$. The lower dimensional vector $\mathcal{Y}^{(\square)} \in \mathbb{R}^n$ captures the most expressive features of the original data $\mathcal{X}^{(\square)}$. The PCA output is then processed by the ICA method and its reduced dimension n is determined based upon the ICA sensitivity analysis.

ICA, which expands on PCA as it considers higher (> 2) order statistics, is used here to derive independent Gabor wavelet features for face recognition. ICA of a random vector seeks a linear transformation that minimizes the statistical dependence between its components [5]. In particular, let $\mathcal{Y} \in \mathbb{R}^n$ be a n dimensional random vector corresponding to the PCA output defined by Eq. 4. The ICA of the random vector \mathcal{Y} factorizes the covariance matrix $\Sigma_{\mathcal{Y}}$ into the following form:

$$\Sigma_{\mathcal{Y}} = F \Sigma F^t \quad (5)$$

where $\Sigma \in \mathbb{R}^{m \times m}$ is diagonal real positive and $F \in \mathbb{R}^{n \times m}$, whose column vectors are orthogonal, transforms the original random vector $\mathcal{Y} \in \mathbb{R}^n$ to a new one $\mathcal{Z} \in \mathbb{R}^m$, where $\mathcal{Y} = F\mathcal{Z}$, such that the m components ($m \leq n$) of the new random vector \mathcal{Z} are independent or “the most independent possible” [5].

The whitening transformation of the ICA derivation can be rearranged as follows [12]:

$$\mathcal{U} = \Sigma^{-1/2} \Sigma^{1/2} \mathcal{Y} \quad (6)$$

The above equation shows that during whitening the eigenvalues appear in the denominator. The trailing eigenvalues, which tend to capture noise as their values are fairly small, cause the whitening step to fit for misleading variations and make the overall method generalize poorly when it is presented with new data. If the whitening step, however, is preceded by a dimensionality reduction procedure (see Eq. 4) and a proper dimensionality is chosen, ICA performance would be enhanced and the computational complexity reduced [12].

3.1 The Probabilistic Reasoning Model for the Independent Gabor Features Method

The novel IGF method applies the independent component analysis on the (lower dimensional) augmented Gabor feature vector. In particular, the augmented Gabor feature vector $\mathcal{X}^{(\square)}$ of an image is first calculated as detailed in Sect. 2. The IGF method determines, then, the dimensionality of the lower dimensional feature space n according to the sensitivity analysis of ICA (Sect. 3) and derives the lower dimensional feature, $\mathcal{Y}^{(\square)}$

(Eq. 4). Finally, the IGF method derives the overall (the combination of the whitening, rotation, and normalization transformations) ICA transformation matrix, F , as defined by Eq. 5. The new feature vector, $\mathcal{Z}^{(\Pi)}$, of the image is thus defined as follows:

$$\mathcal{Y}^{(\Pi)} = F \mathcal{Z}^{(\Pi)} \quad (7)$$

After the extraction of an appropriate set of features, the IGF method applies the Probabilistic Reasoning Model (PRM) [13] for classification. In particular, Let \mathcal{M}_k^0 , $k = 1 \leq 2 \leq \dots \leq L$, be the mean of the training samples for class Π_k after the ICA transformation. The IGF method exploits, then, the following MAP classification rule of the PRM method [13]:

$$\Pi_k = \arg \min_j \frac{\sum_{i=1}^m (z_i^{(\Pi)} - m_{k_i})^2}{\sum_{i=1}^m m_{k_i}^2} \quad (8)$$

where z_i and m_{k_i} , $i = 1 \leq \dots \leq m$, are the components of $\mathcal{Z}^{(\Pi)}$ and \mathcal{M}_k^0 , respectively, and Π_k^2 is estimated by sample variance in the one dimensional ICA space:

$$\Pi_k^2 = \frac{1}{L} \sum_{k=1}^L \frac{1}{N_k \otimes 1} \sum_{j=1}^{N_k} y_{j_i}^{(k)} \otimes m_{k_i} \quad (9)$$

where $y_{j_i}^{(k)}$ is the i -th element of the ICA feature $Y_j^{(k)}$ of the training image that belongs to class Π_k , and N_k is the number of training images available for class Π_k . The MAP classification rule of Eq. 8 thus classifies the image feature vector, $\mathcal{Z}^{(\Pi)}$, as belonging to the class Π_k .

4 Experiments

We assess the feasibility and performance of our novel Independent Gabor Features (IGF) method on the face recognition task, using 600 face images corresponding to 200 subjects from the FERET standard facial database [17]. The effectiveness of the IGF method is shown in terms of both absolute performance indices and comparative performance against some popular face recognition schemes such as the traditional Eigenfaces (PCA) method and Gabor wavelet based classification methods.

For comparison purpose, we use the following nearest neighbor (to the mean) classification rule:

$$\Pi(\mathcal{X} \in \mathcal{M}_k^0) = \min_j \Pi(\mathcal{X} \in \mathcal{M}_j^0) \quad \text{where} \quad \mathcal{X} \in \Pi_k \quad (10)$$

The image feature vector, \mathcal{X} , is classified as belonging to the class of the closest mean, \mathcal{M}_k^0 , using the similarity measure Π . The similarity measures used in our experiments to evaluate the efficiency of different representation and recognition methods include L_1 distance measure, Π_{L_1} , L_2 distance measure, Π_{L_2} , Mahalanobis distance measure, Π_{Md} , and cosine similarity measure, Π_{cos} .

We first implemented the Eigenfaces method [20] on the original images, using the four different similarity measures: L_1 distance measure, Π_{L_1} , L_2 distance measure, Π_{L_2} ,

Table 1. Face recognition performance on the Gabor convolution outputs, using the three different similarity measures.

measure \ representation	$O_{\mu,\nu}^{(16)}$	$\mathcal{X}^{(4)}$	$\mathcal{X}^{(16)}$	$\mathcal{X}^{(64)}$
L_1	76%	76.5%	76.5%	76.5%
L_2	73.5%	72%	72%	72%
cosine	72%	70.5%	70.5%	70%

Mahalanobis distance measure, \square_{Md} , and cosine similarity measure, \square_{cos} . The Mahalanobis distance measure performs better than the L_1 distance measure, followed in order by the L_2 distance measure and the cosine similarity measure. In particular, when 180 features (the specific number of features chosen here facilitates later comparisons with other methods) are used, the recognition rates are 76%, 70.5%, 42.5%, and 38%, accordingly. The reason that the Mahalanobis distance measure performs better than the other similarity measures is that the Mahalanobis distance measure counteracts the fact that L_1 and L_2 distance measures in the PCA space weight preferentially for low frequencies. As the L_2 measure weights more the low frequencies than L_1 does, the L_1 distance measure should perform better than the L_2 distance measure, a conjecture validated by our experiments.

The next series of experiments used the Gabor convolution outputs, $O_{\theta,\square}(z)$, derived in Sect. 4, with the L_1 , L_2 and cosine similarity measures, respectively. For the first set of experiments, we downsampled the Gabor convolution outputs by a factor 16 to reduce the dimensionality and normalized them to unit length, as suggested by Donato et al. [4]. The classification performance using such Gabor outputs is shown in Table 1. The best performance is achieved using the L_1 similarity measure. We have also experimented on the augmented Gabor feature vector $\mathcal{X}^{(\square)}$ as defined by Eq. 3 with three different downsampling factors: $\square = 4, 16$, and 64 , respectively. From the classification performance shown in Table 1, we found that (i) the augmented Gabor feature vector $\mathcal{X}^{(\square)}$ carries quite similar discriminant information to the one used by Donato et al. [4]; and (ii) the performance differences using the three different downsampling factors are not significant. As a result, we choose the downsampling factor 64 for our novel IGF method, since it reduces to a larger extent the dimensionality of the vector space than the other two factors do. (We experimented with other downsampling factors as well. When the downsampling factors are 256 and 1024, the performance is marginally less effective; when the factor is 4096, however, the recognition rate drops drastically.)

The last experiment, performed using the novel Independent Gabor Features (IGF) method described in this paper, shows that the IGF derives independent Gabor features with low dimensionality and enhanced discrimination power. In particular, when 180 features are used by the IGF method, the correct recognition rate is 98.5%.

References

1. M.S. Bartlett, H.M. Lades, and T.J. Sejnowski, "Independent component representations for face recognition," in *Proceedings of the SPIE, Vol 3299: Conference on Human Vision and Electronic Imaging III*, 1998, pp. 528–539.
2. P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.
3. D. Burr, M. Morrone, and D. Spinelli, "Evidence for edge and bar detectors in human vision," *Vision Research*, vol. 29, no. 4, pp. 419–431, 1989.
4. R. Chellappa, C.L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," *Proc. IEEE*, vol. 83, no. 5, pp. 705–740, 1995.
5. P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol. 36, pp. 287–314, 1994.
6. J.G. Daugman, "Complete discrete 2-d Gabor transforms by neural networks for image analysis and compression," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1169–1179, 1988.
7. G. Donato, M.S. Bartlett, J.C. Hager, P. Ekman, and T.J. Sejnowski, "Classifying facial actions," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 974–989, 1999.
8. K. Etemad and R. Chellappa, "Discriminant analysis for recognition of human face images," *J. Opt. Soc. Am. A*, vol. 14, pp. 1724–1733, 1997.
9. D. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *J. Opt. Soc. Amer. A*, vol. 4, no. 12, pp. 2379–2394, 1987.
10. J. Jones and L. Palmer, "An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex," *J. Neurophysiology*, pp. 1233–1258, 1987.
11. M. Lades, J.C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, Wurtz R.P., and W. Konen, "Distortion invariant object recognition in the dynamic link architecture," *IEEE Trans. Computers*, vol. 42, pp. 300–311, 1993.
12. C. Liu and H. Wechsler, "Comparative assessment of independent component analysis (ICA) for face recognition," in *Proc. Second International Conference on Audio- and Video-based Biometric Person Authentication*, Washington D. C., March 22–24, 1999.
13. C. Liu and H. Wechsler, "Robust coding schemes for indexing and retrieval from large face databases," *IEEE Trans. on Image Processing*, vol. 9, no. 1, pp. 132–137, 2000.
14. B. Moghaddam, "Principal manifolds and bayesian subspaces for visual recognition," in *the 7th Int'l Conf. on Computer Vision*, Corfu, Greece, September, 1999.
15. B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696–710, 1997.
16. B.A. Olshausen and D.J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 13, pp. 607–609, 1996.
17. P.J. Phillips, H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," *Image and Vision Computing*, vol. 16, pp. 295–306, 1998.
18. A. Rosenfeld and H. Wechsler, "Pattern recognition: Historical perspective and future directions," *Int. J. Imaging Syst. Technol.*, vol. 11, pp. 101–116, 2000.
19. D. L. Swets and J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Trans. on PAMI*, vol. 18, no. 8, pp. 831–836, 1996.
20. M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 13, no. 1, pp. 71–86, 1991.
21. L. Wiskott, J.M. Fellous, N. Kruger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775–779, 1997.

Face Recognition from 2D and 3D Images

Yingjie Wang, Chin-Seng Chua, and Yeong-Khing Ho

School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore 639798
ecschua@ntu.edu.sg

Abstract. This paper proposes a feature-based face recognition system based on both 3D range data as well as 2D gray-level facial images. Feature points are designed to be robust against changes of facial expressions and viewpoints and are described by Gabor Wavelet filter in the 2D domain and Point Signature in the 3D domain. Localizing feature points in a new image is based on 3D-2D correspondence, their relative position and corresponding Bunch (covering a wide range of possible variations on each feature point). Extracted shape and texture features from these feature points are first projected into their own eigenspace using PCA. In eigenspace, the corresponding shape and texture weight vectors are further integrated through a normalization procedure to form an augmented vector which is used to represent each facial image. For a given test facial image, the best match in the model library is identified according to a similarity function. Experimental results involving 20 persons with different facial expressions and extracted from different viewpoints have demonstrated the efficiency of our algorithm.

1 Introduction

Face recognition finds many potential applications in building/store access control, suspect identification and surveillance. Research activities in this area have gained much attention in the last few years [1], [2], [3]. Almost all existing recognition systems rely on a single type of facial information: 2D intensity (color) images [1], [2] or 3D range data sets [3]. It is evident that range images can represent 3D shape explicitly and can compensate for the lack of depth information in a 2D image. 3D shape is also invariant under the change of color (such as due to a change in a person's facial makeup) or reflectance properties (such as due to a change in the ambient lighting). On the other hand, the variety of gray-level information provided by different persons gives more detailed information for interpreting facial images, albeit its dependence on color and reflectance properties. Thus integrating 2D and 3D sensory information will be a key factor for achieving a significant improvement in performance over systems that rely solely on a single type of sensory data.

For 2D face recognition, Eigenface [1] is a representative method based on Principle Component Analysis (PCA). The basic idea is to code each face by a linear combination of the eigenfaces in a lower dimension. In [1], using the raw

facial image as input to PCA enforces strong restriction in image alignment and illumination and cannot reflect the local variation of face. To a limited extent, Gabor filter responses or other features may be extracted on carefully chosen feature points and using these features as inputs to PCA may help to relax the above limitation. In [2], each face is coded as a face graph whose nodes are described by sets of Gabor components (Jets). A set of Jets referring to one feature point is called a Bunch. The goal of Elastic Bunch Graph Matching on a test image is to find the fiducial points and thus extract from the image a graph that maximizes the similarity. During the localization process coarse to fine approach is used, but is extremely time consuming. To narrow down the search range will be a good way to improve the efficiency of this algorithm.

In our system, corresponding 3D range images and 2D gray-level images can be obtained simultaneously. The main objective of our system is to explore a more efficient feature-based approach by considering both shape and texture information and capitalize on the advantages afforded by using either 2D or 3D information. This paper is organized as follows. Section 2 describes the construction of Bunch for each feature point. These points are localized using both 2D and 3D information. In Section 3, based on the integration of shape and texture features after PCA and a similarity function, the approach for face identification is developed. The efficiency of our algorithm is verified in Section 4 through 3 test cases.

2 Localization of Feature Points

2.1 Definition of Feature Points

In our system, two types of feature points are defined for recognition. One type is in 3D and is described by the Point Signature, while the other is in 2D and is represented by Gabor wavelets. To choose these feature points, computation requirements, distinction, representative and the insensitivity to the expression and viewpoint variations are the main concerns that are considered. We select four 3D feature points and ten 2D feature points, which are shown in Fig.1 (a).

2.2 Gabor Responses

Multi-orientation and multi-scale Gabor Kernels are robust against brightness and contrast variations in the gray image. The Gabor response describes a small patch of gray values in an image $I(\mathbf{x})$ around a given pixel $\mathbf{x} = (x, y)^T$. It is based on a Wavelet transformation, defined as a convolution with a family of Gabor kernels:

$$J_j(\mathbf{x}) = \int I(\mathbf{x}') \psi_j(\mathbf{x} - \mathbf{x}') d^2 \mathbf{x}' \quad (1)$$

$$\psi_j(\mathbf{x}) = \frac{\|\mathbf{k}_j\|^2}{\sigma^2} \exp\left(-\frac{\|\mathbf{k}_j\|^2 \|\mathbf{x}\|^2}{2\sigma^2}\right) [\exp(i\mathbf{k}_j \cdot \mathbf{x}) - \exp(-\frac{\sigma^2}{2})] \quad (2)$$

where $\sigma = 2\pi$, i stands for complex computation and \mathbf{k}_j , the characteristic frequency for each filter, is given by

$$\mathbf{k}_j = \begin{pmatrix} k_v \cos \varphi_u \\ k_v \sin \varphi_u \end{pmatrix}, \quad k_v = 2^{-\frac{v+2}{2}} \pi, \quad \varphi_u = \mu \frac{\pi}{8}. \quad (3)$$

In our research, we employ 5 different discrete frequencies, indexed by $v = 0, \dots, 4$, and 8 orientations indexed by $u = 0, \dots, 7$, where index $j = u + 8v$. A Jet J is defined as the set $\{J_j\}$ of 40 complex coefficients obtained from one image point. It is written as $J_j = a_j \exp(i\phi_j)$, where a_j and ϕ_j correspond to its amplitude and phase respectively. In our system, only the amplitude, a_j , is used for detection and recognition.

2.3 Point Signature

Point Signature (P.S.) is a representation for 3D free-form surfaces [4]. Its main idea is summarized here. For details, the reader may refer to [4]. For a given point p , we place a sphere of radius r , centered at p . The intersection of the sphere with the object surface is a 3D curve C . n_1 is defined as the unit normal vector of the plane fitted through the surface curve C . A new plane P' is defined by translating the fitted plane to the point p in a direction parallel to n_1 . The perpendicular projection of C to P' forms a new planar curve C' with the projection distance of points on C' forming a signed distance profile. We refer to this profile of distances $d(\theta)$ with $0 \leq \theta \leq 360$, as the signature at point p .

In our implementation, we just consider finite sampling angular points with sampling interval $\Delta\theta = 10^\circ$. Hence the distance profile is now represented by a set of discrete values $d(\theta_i)$ for $i = 1, \dots, 36$, $0^\circ \leq \theta_i \leq 360^\circ$. The P.S. of a nose tip is shown in Fig.1 (b). Due to the local property of P.S., in order to identify a 3D feature point uniquely in a range image, in our system, P.S. of a 3×3 mesh centered at this point are considered. Emulating the Jet defined in Gabor filter, we define a 3D Jet for each 3D feature point as a set of $3 \times 3 \times 36$ distance profile.

2.4 Building Bunch and Localizing Feature Points

In order to localize feature points in a new test facial image, as in [2], the Bunch for each feature point is created first. Here the link between each feature point is not considered as in [2]. The 2D position of each feature point in the model image is localized manually first.

The distance between two Jets is defined as $D_3(J, J')$ for 3D points and $D_2(J, J')$ for 2D points, which are given by

$$D_3(J, J') = \frac{1}{M} \sum_{j=1}^M \sqrt{\frac{1}{T} \sum_{i=1}^T [d_j(\theta_i) - d'_j(\theta_i)]^2}, \quad D_2(J, J') = \sqrt{\frac{1}{S} \sum_{i=1}^S (a_i - a'_i)^2} \quad (4)$$

where $d_j(\theta_i)$ is the distance profile for point j , $T = 36$ is the angular sampling number, M is the number of considered points for computing P.S. around point j ;

$S = 40$ is the Gabor filter coefficient (G.C.) size. Based on these two function, we can obtain all the distances between one Jet and all the Jets in the corresponding Bunch. Among these distances, the minimum one is denoted as $D_3^B(J, J^B)$ for 3D points and $D_2^B(J, J^B)$ for 2D points, where

$$D_3^B(J, J^B) = \min_{i=1}^N D_3(J, J^{B_i}), \quad D_2^B(J, J^B) = \min_{i=1}^N D_2(J, J^{B_i}) \quad (5)$$

and N is the number of model images in a Bunch.

Detecting feature points in a new test facial image has roughly two stages. Firstly, based on the average layout and position of each feature point in a face, each 3D feature point in a test range image is localized by choosing the one which minimizes $D_3^B(J, J^B)$ in the search range. Its 2D position is further determined based on 3D-2D correspondence. The detected 3D position of tip of nose is also utilized to estimate head pose approximately. Secondly, with these four 2D known positions, the estimated head pose and the assumption of facial symmetry, each of the rest 2D feature points is now limited in a relatively small region. In this small region, each 2D feature point is further detected by finding the minimum $D_2^B(J, J^B)$. Due to this constrained search range, we actually omit the Stage 1 (Find approximate position) of EGM procedure in [2]. Moreover, pre-construction of bunch graph for different head poses is unnecessary for solving the face recognition problem with different viewpoints.

Some detection results in 2D domain are demonstrated in Fig.1 (c). On the average feature points can be localized accurately despite different viewpoints and expressions. Within the defined feature points, localizations of distinguished feature points such as the tip of nose and inner corner of eyes are more accurate than those of others.

3 Face Identification

Once each feature point P_i is located, its shape feature X_{si} and texture feature X_{ti} are determined. Let X_s and X_t represent the shape and texture feature vector of a facial image. $X_s = [\alpha_1 X_{s1}, \dots, \alpha_4 X_{s4}]^T \in R^{36 \times 9 \times 4}$ consists of P.S. of four meshes centered at 4 detected 3D points. $X_t = [\beta_1 X_{t1}, \dots, \beta_{10} X_{t10}]^T \in R^{40 \times 10}$ consists of G.C. of the localized ten 2D feature points. Here α_i and β_j are the weights corresponding to the i -th 3D and j -th 2D feature point individually. If they all equal to 1, each feature point will have the same importance in the formation of a facial feature vector. Confidence of each feature point is chosen based on its average experimental detection accuracy and the properties of adopted representation method. For example, with P.S. representing each 3D point, nose tip will be more selective than the points on the forehead, so the weighting of nose tip will be chosen greater than that of points on the forehead. With the extracted shape and texture feature vector X_s and X_t of each training facial image F_i , their covariance matrices C_s and C_t are derived through:

$$C = \frac{1}{N} \left[\sum_{i=1}^N (F_i - F)(F_i - F)^T \right], \quad F = \frac{1}{N} \sum_{i=1}^N F_i \quad (6)$$

where N is the training set number, F is the average facial image. We chose only m eigenvectors Φ_1, \dots, Φ_m to represent the original image space n where $m < n$. For a new facial image F' , it is projected into this subspace by a simple operation: $\omega_k = \Phi_k^T(F' - F)$. These weights form a vector $\Omega = [\omega_1, \omega_2, \dots, \omega_m]^T$ to represent F' . When the shape feature weight vector Ω_s and texture feature weight vector Ω_t of F' are obtained, they are integrated to form an augmented feature vector Ω_{st} by Eq.(7) after normalization. In our research this final weight vector Ω_{st} is used to register each training and test facial image.

$$\Omega_{st} = (\Omega_s^T / \|\Omega_s\|, \Omega_t^T / \|\Omega_t\|)^T \quad (7)$$

$$S(\Omega_{st}, \Omega'_{st}) = \frac{\sum_j \omega_j \omega'_j}{\sqrt{\sum_j \omega_j^2 \sum_j \omega'^2_j}} \quad (8)$$

Since the shape and texture elements in Ω_{st} belong to two different eigenspaces, Mahalanobis Distance is unavailable here. In order to compare two facial images Ω_{st} and Ω'_{st} , we define a similarity function given by Eq.(8). All the similarities between a new test image and the model images are calculated based on this function. The best-matched model facial image is the one that has the greatest similarity with the test image.

4 Experiments

In order to testify the efficiency of this method, experiment was processed using facial images of 20 individuals in our laboratory. The METRICOR-3D Precise Shape Measurement System with CCD cameras and fame grabber are used to capture range images and the corresponding gray-level images. Correspondence between these two kinds of images can be known from the measured image data. After filtering and interpolation preprocesses, each range image is used to determine the corresponding face region in the 2D gray image. In our system, each person provides 6 facial images with viewpoint and facial expression variations. In our system, three images of each person are used as training images to form the eigenspace, and one of these 3 images is used to build Bunch. The rest facial images are in turn taken as test set for recognition. Sample images involving 8 individuals are shown in Fig.1 (d) and (e).

After localizations of the defined feature points as described in Section 2, we construct a $36 \times 9 \times 4$ -dimension shape vector and a 40×10 -dimension texture vector for each facial image. These two vectors are further transformed into the eigenspace using PCA to form the weight vector Ω_{st} . For comparison we conducted 3 test cases with different types of features and different numbers of chosen principle components. Fig.1 (f) shows the recognition rate versus eigenspace dimension with chosen features as P.S., G.C. and P.S. + G.C.. The recognition results show clearly that the highest recognition rate is achieved by considering both of the shape and texture features and using more principle components to construct eigenspace. This is reasonable because when a facial image is represented with more information, it will be more discriminating and then easier to discern from others.

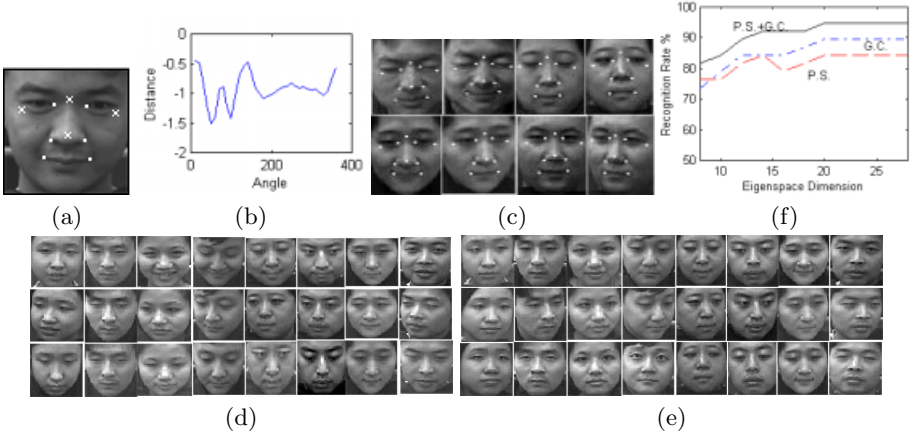


Fig. 1. (a) Selection of ten 2D points (shown by "." and "x") and four 3D points (shown by "x"); (b) P.S. of one nose tip; (c) Some localization results; (d) Training sample images; (e) Test sample images; (f) Recognition rate vs. eigenspace dimension.

5 Conclusion

We present a face recognition algorithm using corresponding range and gray-level facial images. Experiment results have demonstrated the promising performance of our algorithm. For a larger face database, due to the discriminating representation by integrating the shape and texture information and the robustness of the chosen feature in our system, our algorithm will not deteriorate, which will be evaluated in our future work. Furthermore, other classifiers such as Support Vector Machine will be adopted in our system.

References

1. Turk, M. and Pentland, A.: Eigenface for Recognition. *Journal of Cognitive Neuroscience* **3** (1991) 72–86.
2. Wiskott, L., Fellous, J.M., Kuiger, N., and von der Malsburg, C.: Face Recognition by Elastic Bunch Graph Matching. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **7** (1997) 775–779.
3. Chua, C.S., Han, F., and Ho, Y.K.: 3D Human Face Recognition Using Point Signature. *Fourth IEEE International Conference on Automatic Face and Gesture Recognition* (2000) 233–238.
4. Chua, C.S. and Jarvis, R.: Point Signature: A New Representation for 3D Object Recognition. *International Journal on Computer Vision* **1** (1997) 63–85.
5. Liu, C.J. and Wechsler, H.: Face Recognition Using Shape and Texture. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **1** (1999) 598–603.

Face Recognition Using Support Vector Machines with the Feature Set Extracted by Genetic Algorithms

Kyunghee Lee¹, Yongwha Chung¹, and Hyeran Byun²

¹ Electronics and Telecommunications Research Institute
161 Gajung-Dong, Yuseong-Gu, Daejeon, 305-350, Korea
{uniromi, ywchung}@etri.re.kr

² Yonsei University,
134 Sinchon-Dong, Seodaemun-Gu, Seoul, 120-749, Korea
hrbyun@aipiri.yonsei.ac.kr

Abstract. Face recognition problem is challenging because face images can vary considerably in terms of facial expressions, 3D orientation, lighting conditions, hair styles, and so on. This paper proposes a method of face recognition by using support vector machines with the feature set extracted by genetic algorithms. By selecting the feature set that has superior performance in recognizing faces, the use of unnecessary information of the faces can be avoided and the memory requirement can be decreased significantly. Also, by using a tuning data set in the computation of the evaluation function, the feature set which is less dependent on illumination and expression can be selected. The experimental results show that the proposed method can provide superior performance than the previous method in terms of accuracy and memory requirement.

1 Introduction

Face is one of the most acceptable biometrics because it is one of the most common methods of identification which humans use in their visual interactions. In addition, the method of acquiring face images is non-intrusive. However, it is difficult to develop an automatic face recognition system, while people can easily recognize familiar human faces. It is because face images can vary considerably in terms of facial expressions, 3D orientation, lighting conditions, hair styles, and so on.

Two primary approaches to the identification based on face recognition are the *holistic* (or *transform*) approach and the *analytic* (or *attribute-based*) approach [1]. In the holistic approach, the universe of face image domain is represented using a set of orthonormal basis vectors. Currently, the most popular basis vectors are eigenfaces [2]. Each eigenface is derived from the covariance analysis of the face image population. Two faces are considered to be identical if they are sufficiently close in the eigenface feature space. A number of variants of such an approach exist. Template matching-based face recognition systems are also classified into this approach.

In the analytic approach, facial attributes like nose, eyes, etc. are extracted from the face image and the invariance of geometric properties among the face landmark

features is used for recognizing features [3]. This approach has characteristics of high-speed and low-memory requirement, while the selection and extraction of features are difficult. Many previous works using this approach have been reported. For instance, recognition accuracy of 90% could be achieved by using geometric features [4]. Hybrid approaches combining both holistic and analytic have also been reported [1].

Recently, face detection and recognition systems using *Support Vector Machine* (SVM) binary classifiers [5] have been proposed. For example, methods for detecting face images [6], face pose discrimination [7], and recognizing face images [8-11] have been proposed using SVMs. The SVM-based performance has also been compared with the Principal Component Analysis-based performance [12]. For the commercial purposes, however, the performance of the SVM-based method needs to be improved further.

On the contrary, a Genetic Algorithm (GA)-based representation transformation has been developed recently [13]. It can select and create appropriate features in order to represent a problem suitably. Also, the visual routine for eye detection has been developed by integrating the decision tree learning and the GA [14]. We apply the GA to the face recognition.

In this paper, we propose a method that uses the feature set extracted by GAs as an input vector to an SVM. In the previous SVM-based methods, the whole image is used as an input vector so that unnecessary information as well as necessary information is included in creating a SVM. In our method, the use of unnecessary information of the faces can be avoided by selecting the feature set which has superior performance in recognizing faces. The memory requirement can be decreased significantly. Also, by using a tuning data set in the computation of the GAs evaluation function, the feature set which is less dependent on illumination and expression can be selected. The experimental results show that our feature set-based method can provide superior performance than the whole image-based SVM method.

The organization of the paper is as follows. Background is given in Section 2. In Section 3, the proposed method for recognizing faces is explained. Experimental results are shown in Section 4, and concluding remarks are made in Section 5.

2 Background

For the purpose of completeness, both SVM and GAs are explained briefly. Details of SVM and GAs can be found in [5,15]. Support Vector Machine (SVM) is a binary classification method that finds the optimal linear decision surface based on the concept of structural risk minimization. The decision surface is a weighted combination of elements of the training set. These elements are called *support vectors* and characterize the boundary between the two classes.

Genetic Algorithm (GA)s [15] are adaptive and robust computational procedures modeled on the mechanics of natural genetic systems. GAs typically maintain a constant-sized population of individuals that represent samples of the intended search

space. Each individual is evaluated on the basis of its overall fitness with respect to the given application domain. New individuals are produced by selecting high-performing individuals to produce offspring that retain many of the features of their parents, resulting in an evolving population.

3 Proposed Method

In our method, the GA is used to explore the space of all subsets of the given feature list. A preference is given to those subsets of the features which can achieve the best classification performance using small dimensionality feature sets. Each of the selected feature subsets is evaluated using an SVM. This process is iterated along the evolutionary lines until the best feature subset is found.

3.1 Selected Feature Set Using Genetic Algorithm

Each chromosome can be represented as a fixed-length binary string standing for some subset of the original feature list. Each bit of the chromosome represents whether the corresponding feature is selected or not. 1 in each bit means the corresponding feature is selected, whereas 0 means it is not selected. During the evolution, the simple cross-over operator and the elitist strategy [15] are used.

3.2 Face Recognition Using Support Vector Machine

In this paper, a SVM is generated by using a polynomial kernel with the selected features as input vectors. To evaluate the performance of the SVM generated, the classifier performance of the SVM is measured using a tuning data set. Then, the system (the SVM) evolves using this classifier performance as an evaluation function. After the evolution, the SVM becomes a face recognition system with the features representing most prominent chromosomes as input vectors.

The overall structure of the face recognition system proposed in this paper is shown in Fig. 1. The processing begins with the preprocessing and the feature extraction steps for each image used for training. To reduce the illumination affect, the histogram equalization is used in the preprocessing step. Also, to reduce duplicate computation during the GA step, the feature set used for the feature selection is precomputed. The feature set is derived by computing the averages of the pixels within the subwindows before and after the Sobel edge operator respectively.

The next step is the evolution step using GAs. Each subset of the feature set is represented as a chromosome, and the evaluation function is applied to each chromosome. To computer the fitness value of the chromosome, a SVM is generated first with the input vector which corresponds to the features representing each chromosome from the training images. Then, with the SVM generated, the classification rates of the tuning images are used as the fitness values. The algorithm terminates when either the

fitness value of the superior chromosome becomes 1.0 or the number of generations becomes a specific value.

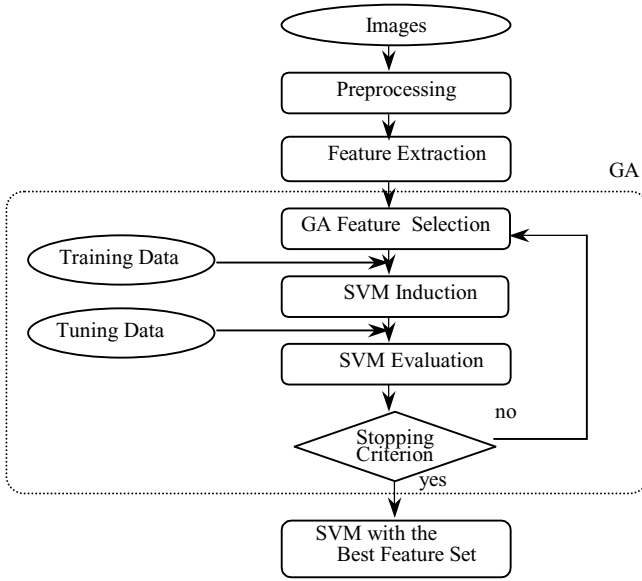


Fig. 1. A Face Recognition System Proposed.

4 Experimental Results and Analysis

To evaluate the proposed method, two sets of experiments were conducted for FRR and FAR respectively. The code was implemented in Visual C++ and run on an 800MHz Pentium III PC. The test database we used is the Yale Face database [16] which is widely used in the face recognition community. It consists of 11 images per 15 people, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. In this paper, each data in the database was used for one of the three purposes as follows:

- ✂ *Training data set* to generate a SVM. For each person, 3 images of himself and 3 images of other 5 people were used for training.
- ✂ *Tuning data set* to compute evaluation function. For each person, 3 images of himself and 3 images of other 5 people were used for tuning.
- ✂ *Test data set* to evaluate performance after training. For each person, 5 images of himself and 124 images of other 14 people were used for testing. Among these 124 images, 44 images were unknown people's images at either the training or the tuning.

The overall feature set from the images of size 64 64 was computed as follows. Each image was scanned with the window of size 8 8 with a 4-pixel overlap, generating 225 windows. Averages of the pixels in each 225 window before and after the Sobel operator generated 450 features. Then, for each person, specific features which can recognize himself effectively were selected from this overall feature set. The selection can be represented as 1(select) or 0(no select) in the 450-bit chromosome.

For each person of the test database, the proposed method showed 5.3% False Reject Rate(FRR) and 2.2% False Accept Rate(FAR) on the average. Especially, 1.8% FAR could be achieved with different people s images which were known at the training/tuning. Even with different people s images which were unknown at the training/tuning, 3.0% FAR could be achieved.

To evaluate the effectiveness of the proposed method, the same experiments were conducted with the previous SVM method. It used the whole input image pixels, instead of the selected features. As shown in Fig. 2 and 3, the previous method showed 18.7% FRR and 4.2% FAR on the average. Also, the number of features in the previous method was 4096 from the images of size 64 64, whereas the numbers of features in the proposed method were between 209 and 231. Especially, this memory saving is very effective in large-scale face identification systems.

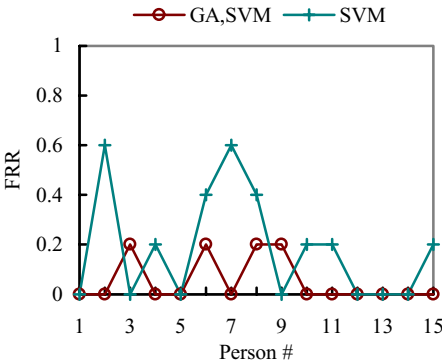


Fig. 2. FRR of the Previous and the Proposed Methods with Yale Database.

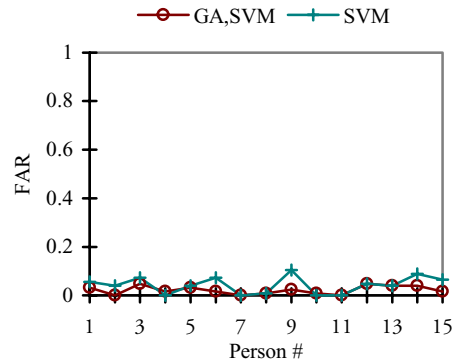


Fig. 3. FAR of the Previous and the Proposed Methods with Yale Database.

5 Concluding Remarks

This paper proposed a method of face recognition by integrating GA into SVM. Compared to the previous methods which use the whole image as an input vector, this method uses the feature set extracted by GA as an input vector. Our proposed method has several advantages. By selecting the feature set which has superior performance in recognizing faces, the use of unnecessary information of the faces can be avoided and

the memory requirement can be decreased significantly. Also, by using a tuning data set in the computation of the GA evaluation function, the feature set which is less dependent on illumination and expression can be selected.

For the evaluation purpose, we selected the Yale Face database having diverse face images in terms of illumination and expression, as our test set. Our experimental results showed that FRR of our feature set-based SVM method was 5.3% versus 18.7% for the whole image-based SVM method. FAR of our method was 2.2%, whereas it was 4.2% for the previous method. We are currently conducting further evaluation of the proposed method using more face images. We expect better performance can be achieved by training a SVM with large number of face images. Also, the small memory requirement of the proposed method makes it applicable to either large-scale face identification systems or memory-constrained smart card systems.

References

1. K. Lam and H. Yan. An Analytic-to-Holistic Approach for Face Recognition based on a Single Frontal View. *IEEE Tr. on PAMI*, Vol. 29, No. 7, pp. 673-686, 1998.
2. M. Turk and A. Pentland. Face Recognition using Eigenfaces. *Proc. of CVPR*, pp. 586-591, 1991.
3. C. Wu and J. Huang. Human Face Profile Recognition by Computer. *Pattern Recognition*, Vol. 23, No. 3/4, pp. 255-259, 1990.
4. R. Brunelli and T. Poggio. Face Recognition: Features versus Templates. *IEEE Tr. on PAMI*, Vol. 15, No. 10, pp. 1042-1052, 1993.
5. V. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, New York, 1998.
6. E. Osuna, R. Feund, and F. Girosi. Training Support Vector Machines: an Application to Face Detection. *Proc. of CVPR*, pp.130-136, 1997.
7. J. Huang, X. Shao, and H. Wechsler. Face Pose Discrimination using Support Vector Machines. *Proc. of ICPR*, pp. 154-156, 1998.
8. G. Guo, S. Li, and K. Chan. Face Recognition by Support Vector Machines. *Proc. of FGR*, pp. 196-201, 2000.
9. K. Jonsson, J. Matas, J. Kittler, and Y. Li. Learning Support Vectors for Face Verification and Recognition. *Proc. of FGR*, pp. 208-213, 2000.
10. Y. Li, S. Gong, and H. Liddell. Support Vector Regression and Classification based Multi-View Face Detection and Recognition. *Proc. of FGR*, pp. 300-305, 2000.
11. B. Moghaddam, M. Yang. Gender Classification with Support Vector Machines. *Proc. of FGR*, pp. 306-311, 2000.
12. P. Phillips. Support Vector Machines Applied to Face Recognition. *Technical Report*, NIST, 1999.
13. H. Vafaie and K. DeJong. Feature Space Transformation Using Genetic Algorithms. *IEEE Intelligent Systems*, March/April, pp.57-65, 1998.
14. J. Bala, K. DeJong, J. Huang, H. Vafaie, and H. Wechsler. Visual Routine for Eye Detection Using Hybrid Genetic Architectures. *Proc. of ICPR*, pp. 606-610, 1996.
15. David E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
16. Yale Face Database. <http://cvc.yale.edu/projects/yalefaces/yalefaces.html>.

Comparative Performance Evaluation of Gray-Scale and Color Information for Face Recognition Tasks

Srinivas Gutta¹, Jeffrey Huang², Chengjun Liu,³ and Harry Wechsler⁴

¹ Philips Research, 345 Scarborough Rd., Briarcliff Manor, NY 10510, USA
Srinivas.Gutta@philips.com

² Dept. of Computer and Info. Science, Indiana University - Purdue University
723 West Michigan St. SL 280C, Indianapolis, IN 46202, USA
huang@cs.iupui.edu

³ Department of Computer Science, 318 Computer Center Building, University of Missouri -
St. Louis, 8001 Natural Bridge Road, St. Louis, Missouri 63121-4499, USA
cliu@cs.umsl.edu

⁴ Department of Computer Science, George Mason University, Fairfax, VA 22030, USA
wechsler@cs.gmu.edu

Abstract. This paper assesses the usefulness of color information for face recognition tasks. Experimental results using the FERET database show that color information improves performance for detecting and locating eyes and faces, respectively, and that there is no significant difference in recognition accuracy between full color and gray-scale face imagery. Our experiments have also shown that the eigenvectors generated by the red channel lead to improved performance against the eigenvectors generated from all the other monochromatic channels. The probable reason for this observation is that in the near infrared portion of the electro-magnetic spectrum, the face is least sensitive to changes in illumination. As a consequence it seems that the color space as a whole does not improve performance on face recognition but that when one considers monochrome channels on their own the red channel could benefit both learning the eigenspace and serving as input to project on it.

1 Introduction

There are several related face recognition tasks, among them: (i) location of a pattern as a face (in the crowd) and estimation of its pose, (ii) detection of facial landmarks, such as the eyes for normalization purposes, and (iii) face recognition - identification and/or verification [1]. A face recognition system starts by locating facial patterns and detecting a set of facial features, for example, the eyes. Using the estimated location of the facial features, the face is normalized for geometrical and illumination variations. The face is then identified using a classification algorithm. Recent human studies [2] suggest that face recognition is mediated by a knowledge of 3D structure of the face and that if the hue of a color image is changed without affecting its luminance, the perceived 3D structure will not alter. It is relevant to check out, if in the context of automatic face recognition, a similar conjecture holds and therefore one should not expect any significant advantage from the use of color over gray-scale

information for recognition accuracy. Towards that end, we have comparatively assessed the usefulness of gray-scale and color information for each of the key face recognition tasks listed above. The image data comes from the standard FERET database [3]. Our findings indicate that using color leads to significant improvements for face location and eye detection, while as expected no significant difference was observed for face recognition.

2 Face Location

To explore the extent, to which color information is helpful with face location, we have performed two experiments for locating the face using (i) color and (ii) gray scale images, respectively. The approach used for face location involves two stages: (i) windows classification and (ii) post processing. The first stage labels 4×4 windows as face or non-face cases using decision trees (DT). The DT algorithm used is Quinlan's C4.5 [4]. The DT are induced ('learned') from both positive ('face') and negative ('non-face') examples expressed in terms of features such as the entropy and the mean with / without Laplacian preprocessing. The same types of features are used for both gray-scale and color images. The labeled output from the first stage is post processed using horizontal and vertical projections to locate the boundaries defining the face boxes. The projections count the number of 4×4 windows labeled as face cases and a large count along either direction suggests the location of a face.

For the color data set, the color space $\{R, G, B\}$ is normalized to the $\{r, g, b\}$ space, where $r = R/(R+G+B)$, $g = G/(R+G+B)$ and $b = B/(R+G+B)$. Each 4×4 window and its corresponding four 2×2 quadrants are processed to yield the features required for learning from examples and the derivation of the DT. Entropy and mean features are derived using original image data or Laplacian preprocessed image data from each 4×4 window to yield 36 features and 12 feature values for color and gray-scale images, respectively. Once the features values become available learning from examples will induce the DT. As inductive learning requires symbolic data, each one of the positive (+) examples corresponding to face regions is tagged FACE, while the negative (-) or counter-positive examples corresponding to non-face regions are tagged as NON-FACE. The input to C4.5 consists of a string of such learning events, each event given as a vector of 36 or 12 attribute values, respectively.

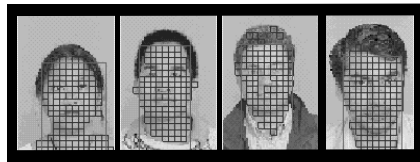


Fig. 1. Examples of Face Boxes.

Both the color and its corresponding gray-scale data set consist of 200 images at a resolution of 64×96 , with each color or gray-scale channel encoded using 8 bits. Among the 200 images, 20 images are used to generate training examples while the remaining 180 images are used as test cases. From each of the 20 training images, a

total of 384 windows of dimension 4 x 4 (some boxes corresponding to the face and others corresponding to the non-face regions) were manually labeled. 7680 windows (+/- examples) consisting of 36 or 12 feature values for each window were used to train the DT. The accuracy rate on face location is based on visual inspection to determine if the box obtained includes both eyes, nose, and mouth, and that the top side of the box is below the hairline. Examples of the box faces found on gray scale images are shown in Fig. 1. The accuracy rates for locating the correct face box are 85.5 % and 90.6% for gray-scale and color images, respectively, a difference found to provide a significant advantage for using color.

3 Eye Detection

We explore the extent to which color information helps with eye detection when compared to using gray levels only. This evaluation is made using a Radial Basis Function (RBF) [5] network. The reasons behind using RBF are its ability for clustering similar images before classifying them. The RBF input consists of **n** normalized eye images fed to the network as 1D vectors for gray scale images. For the color images the RBF input consists of **n** x 3 (R G B) fed to the network as interlaced 1D vectors.

		True Response	
		Class 1 (Eye)	Class 2 (Non-eye)
Predicted	Class 1 (Eye)	88.5 % (Test1)	9.32 % (Test1)
		83.25 % (Test2)	10.44 % (Test2)
		94.75 % (Test3)	14.55 % (Test3)
Response	Class 2 (Non-eye)	11.5 % (Test1)	90.68 % (Test1)
		16.75 % (Test2)	89.56 % (Test2)
		5.25 % (Test3)	85.45 % (Test3)

Table 1. Confusion Map for Tests 1,2,3 Gray Scale.

Experiments were run on a total of 600 face images corresponding to 200 subjects with a resolution of 150 x 130 and encoded using 8 bits. Each subject's image appears as a set of three images (**fa** - frontal gray scale image taken under normal (incandescent) lighting conditions; **fb** - frontal gray scale image again taken under normal lighting conditions but taken after 10 - 15 minutes, and **fc** - frontal gray scale image taken afterwards using fluorescent lighting conditions. The 600 images were divided into sets of 100 and 500 images for training and testing, respectively. The training images were taken only from the **fa** set. From each of the 100 training images, a total of four boxes of dimension 24 x 36 (two boxes corresponding to the left and right eyes, and two boxes corresponding to the non-eye regions) were cut out and used as training examples for the RBF network (two output classes - eye and non-eye). The trained RBF network is tested on the 100 **fa** images not in the training set (Test1), 200 **fb** (Test2) and 200 **fc** (Test 3) images. Testing is performed by scanning a window of size 24 x 36 with an overlap window of 12 x 18 across the entire face image. The confusion maps obtained for the three tests for the gray scale images and

color images are shown in Table 1 and Table 2, and one finds out that the use of color leads to significant improvements over the use of gray-scale only.

		True Response	
		Class 1 (Eye)	Class 2 (Non-eye)
Predicted	Class 1 (Eye)	93 % (Test1)	9.58 % (Test1)
		90.75 % (Test2)	11.26 % (Test2)
		95.5 % (Test3)	15.86 % (Test3)
Response	Class 2 (Non-eye)	7 % (Test1)	90.42 % (Test1)
		9.25 % (Test2)	88.74 % (Test2)
		4.5 % (Test3)	84.14 % (Test3)

Table 2. Confusion Map for Tests 1,2,3 Color.

4 Face Recognition

We now assess the usefulness of color information for face recognition. Towards that end, we use the Principal Component Analysis (PCA) method. Sirovich and Kirby [6] showed that any particular face can be economically represented along the eigenpictures coordinate space, and that any face can be approximately reconstructed by using just a small collection of eigenpictures and the corresponding projections ('coefficients') along each eigenpicture. Since eigenpictures are fairly good in representing face images, one can also consider using the projections along them as classification features to recognize faces.

image sets	image ID	image shots	# images
B ₁	1 to 65	fa and fb	130
B ₂	66 to 130	fa and fb	130
B ₃	131 to 195	fa and fb	130
B ₄	1 to 65	fc	65
B ₅	66 to 130	fc	65
B ₆	131 to 195	fc	65

Table 3. Partition of Images.

Experiments were conducted using 585 gray level and 585 color facial images, corresponding to 195 subjects, from the FERET database. Each subject has three images **fa**, **fb**, and **fc** (see Sect. 3). The images are cropped to a 64 x 96 size, the eye coordinates were manually detected, and the face images are normalized to a fixed interocular distance. The comparative assessment on the usefulness of color vs gray-scale information for face recognition using PCA is carried out along two dimensions. Specifically, cross validation (CV) experiments are carried out to assess the comparative performance (i) when the training data is acquired using incandescent

illumination (yellow tint) while tested on data acquired using fluorescent illumination (greenish tint) Test 1 and (ii) when the origin of the prototype eigenfaces used during testing comes from a disjoint set of face images Test 2 .

The 1170 gray level and color images were divided into 12 disjoint sets; Table 3 shows the 6 sets for gray level images. The other 6 sets for color images ($C_1, C_2, C_3, C_4, C_5, C_6$) use the same partition. Six (6) groups of cross validation (CV) experiments were designed to assess the comparative performance of gray-scale vs color for face recognition. As an example, during TEST 1 we use B_1 and B_2 as training sets to compute principal components (PCs), and the average images (mean values of \mathbf{fa} and \mathbf{fb} of the subjects) are then projected onto those PCs to derive prototypes. During testing, the images from the sets B_4 and B_5 are projected along PCs, and recognition is carried out. During TEST 2, both the images (from set B_3) used to compute prototypes and the images for testing (from set B_6) are projected onto the PCs (derived from image sets B_1 and B_2) for face recognition. Table 4 below shows the average performances when gray-scale and color images were used. The comparative results indicate that there is not a significant difference in recognition performance between color and gray-scale imagery. Furthermore, from a computational point of view, using color would increase both the computation time and the memory requirements.

	TEST 1 Performance		TEST 2 Performance	
Image Type	Gray-Scale	Color	Gray-Scale	Color
Average Performance	93.08 %	91.28 %	92.31 %	91.79 %

Table 4. Average Performance for Face Recognition Gray-Scale vs Color.

Another method for measuring the effects of using color is to generate eigenvectors from each channel independently, and then to use different channels for the gallery and probe set in order to generate the corresponding eigenfaces. For example, one can generate the eigenvectors from the red channel, and use then the green channel for the gallery and probe sets. We generated eigenvectors from the color channels - \mathbf{R} , \mathbf{G} , and \mathbf{B} and from the gray-scale - \mathbf{S} . Each entry C_i in the matrix E denotes channel i for learning the eigenvector space and then projecting the channel C along the eigenspace generated by i .

The following three types of experiments were run. The first ('diagonal') experiment, compares the performance on an individual basis for each element along the diagonal of matrix E . In other words, the channel used to learn the eigenfaces and the channel used to describe both the gallery and the probes are one and the same. The best result is obtained for the case when the channel used is red. The next ('horizontal') experiment fuses information from all the color channels for eigenfaces learned from one channel only and compares the performance against the case when gray-scale is projected along the same eigenfaces. As an example, the top row of matrix E suggests a horizontal experiment where both the color channels and the gray-scale channel are projected along eigenfaces learned using the red channel only; the last row requires that the projections are carried out along eigenfaces learned from gray-scale images. The horizontal experiment shows that the best performance is

obtained when the eigenfaces are learned from the red channel. The last experiment would use the same channel for both learning the eigenfaces and for the gallery and probe sets. One compares now $\{\mathbf{R}_r, \mathbf{G}_g, \mathbf{B}_b\}$ and \mathbf{S}_s , and finds no significant difference in results.

The above experiments show that the eigenvectors generated by the red channel lead to improved performance against the eigenvectors generated from all the other channels for all galleries and probe sets. It appears that the best monochrome channel might be the red channel. The reason for this observation is that in the near infrared portion of the electro-magnetic spectrum, the face is least sensitive to changes in illumination. As a consequence it seems that the color space as a whole does not improve performance on face recognition but that when one considers monochrome channels on their own the red channel should be preferred.

5 Conclusions

To assess the usefulness of color information for biometrics we performed a comparative study of color and gray-scale imagery for face recognition tasks. Experimental results using the FERET database show that color information improves performance for detecting and locating eyes and faces, respectively, and that there is not a significant difference in recognition accuracy between color and gray-scale imagery. Our experiments also seem to indicate that the color space as a whole does not improve performance on face recognition but that when one considers monochrome channels on their own the red channel could benefit both learning the eigenspace and serving as input for to project on it. Further experiments are needed to validate this conjecture, possibly using additional representational subspace methods, such as Fisher Linear Discriminant (FLD) and ICA (Independent Component Analysis).

References

1. Samal, A. and Iyengar, P.: Automatic Recognition and Analysis of Human Faces and Facial Expressions A Survey. *Pattern Recognition* **25** (1992) 65-77.
2. Kemp, R, Pike, G., White, P., and Musselman, A.: Perception and Recognition of Normal and Negative Faces The Role of Shape from Shading and Pigmentation Cues. *Perception* **25** (1996) 37-52.
3. Philips P. J., Wechsler, H., Huang, J., and Rauss, P.: The FERET Database and Evaluation Procedure for Face Recognition Algorithms. *J. Image Vision Comp.* **16(5)** (1998) 295-306.
4. Quinlan, J. R.: C4.5 - Programs for Machine Learning, Morgan Kaufmann (1993).
5. Haykin, S.: *Neural Networks*, Maxmillan Publishing Company (1999).
6. Kirby, M. and Sirovich, L.: Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces. *IEEE Trans. PAMI. Intel.* **12(1)** 1990 103-108.

Evidence on Skill Differences of Women and Men Concerning Face Recognition

Josef Bigun, Kwok-wai Choy, and Henrik Olsson

Halmstad University, Box 823, S-301 18 Halmstad, Sweden

Abstract. We present a cognitive study regarding face recognition skills of women and men. The results reveal that there are in the average sizeable skill differences between women and men in human face recognition. The women had higher correct answer frequencies than men in all face recognition questions they answered. In difficult questions, those which had fewer correct answers than other questions, the performance of the best skilled women were remarkably higher than the best skilled men. The lack of caricature type information (high spatial frequencies) hampers the recognition task significantly more than the lack of silhouette and shading (low spatial frequencies) information, according to our findings. Furthermore, the results confirmed the previous findings that hair style and facial expressions degrades the face recognition performance of humans significantly. The reported results concern 1838 individuals and the study was effectuated by means of Internet.

1 Introduction

The seminal study of Bruce and Young [2] aims to develop a theoretical model and a set of terms for understanding and discussing how we recognize familiar faces, and the relationship between recognition and other aspects of face processing. Faw [6], who, along with Bruce and Young [3], we refer to for an extensive review on human skills of face recognition, compares 112 human studies about face recognition and investigates the need for a standardization.

Hassing et. al. [7] studied the relative importance of age, gender and education on episodic memory functioning in a population-based sample of healthy individuals, between 90 and 100 years of age. The results, that do not quantify the male and female skills, suggest that education, age and gender has no effect in face recognition skills in this age category. Other age categories were not studied. Twenty test pictures were shown to subjects during 6 seconds to subjects who should later recognize these among 20 distractor pictures. The observed face recognition skills of the subjects were modulated by the memory skills.

Bruce and Beard [8], studied female African American and Caucasian Americans' ability to recognize female faces of their own racial group and/or another racial group. They report on two experiments. In the Experiment 1, participant subjects saw either African American or Caucasian American faces; in Experiment 2, all participants saw faces of both races. There was no evidence of cross-racial bias in Experiment 1. Experiment 2 revealed some evidence of cross-racial

bias, in which Caucasian Americans performed more poorly and made more errors in recognition of African American faces than the African Americans did on Caucasian Americans.

The Priming effects in children's face recognition are studied by Ellis et.al. [5]. The subjects (children and young adults) were told to describe either face expression or gender. Subsequently some familiar person's pictures among unfamiliar persons' pictures were shown for a judgment. There were three age categories (five-, eight-, eleven-year old). One half were assigned to judge expression and the other half to judge gender. The reaction time was estimated with the computer in order to see how fast or good the face recognition was. The experiment indicates that participants of five years show the same reaction time as older children in face recognition. In this experiment the pictures were the same on each occasion. In the second experiment they also showed a different view of the face and the result concerning the reaction time was the same. Consequently, there are not developmental changes in the implicit memory. According to this study there are developmental changes in the explicit memory.

We present below original evidence that uncovers the significant skill differences between the genders as they apply to face recognition based on a large cognitive study. We also present evidence confirming known distractor factors presented in other studies.

2 Method

The experiments were conducted in a spatially distributed manner roughly during one month by using the Internet and web pages.

2.1 The Host Program

Upon starting the test, a Java program, which we will here refer to as the *host program*, specifically written for the present study, walked the subject through the various web pages, step by step until all answers were registered. The host program had a number of questions attempting to quantify some face recognition skills that will be described in detail further below. In the design of the host program, special attention was given to that the subjects were prevented to give more than one answer to a question and that once a question was answered they were only allowed to go to the next question by pushing the "Next" button. Consequently, the questions had to be answered one after the other but the subject decided without constraint on time when to fill the answer and thereby move on to the next question.

The following compulsory information was gathered from every subject who completed the test:

1. *recognition data* consisting of the answers of the subject to the face recognition questions,
2. *subject data* consisting of the physical (gender and age) and geographical data (city type and continent) of the subject.



Fig. 1. The test object (left-top) of Question 1, to be identified among the 10 objects underneath. On the right, the analogous data for Question 2 are given.



Fig. 2. The test object (left-top) of Question 3, to be identified among the 10 objects underneath. On the right, the analogous data for Question 4 are given.

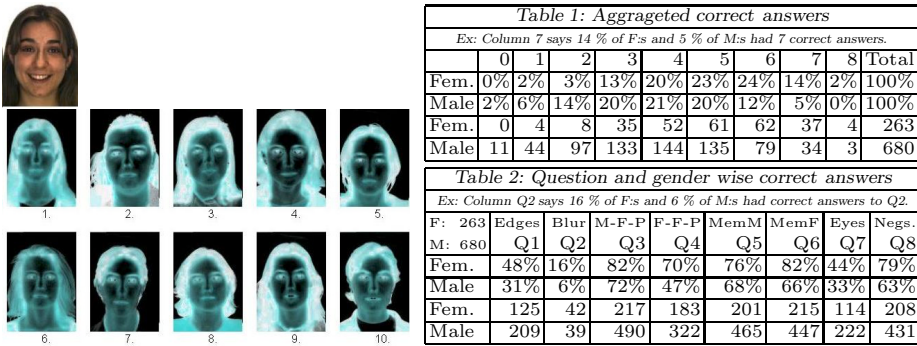


Fig. 3. The images used in Question 8. Tables 1 and 2 illustrate the aggregated CAs and detailed answer statistics.

Terminating the host program by closing the window was of course possible but no registration of the data took place at Halmstad University server in that case. In addition to the compulsory information the following voluntary information was gathered from the subjects:

- name, address, and E-mail.

2.2 The Face Recognition Questions

The test consisted of 8 questions, (Q1,...Q8). The task was to identify the picture of an object person among a set of 10 pictures of persons, object set. In Q1-Q4, and Q8 the test object was shown on the top of the object set simultaneously, in the same page. The Q5-Q7 were similar to other questions except that they included a memory task i.e. the test object was shown in its own page, after the close of which by the test subject, a page containing only the object set was presented.

The subjects were informed before the start of the test object image and the image to be found in the object set were taken at two different occasions, i.e. they were told that, the two images could differ significantly in hair style, glasses, expression of the face, facial hair, clothing, ... etc due to the natural changes in appearance that occur upon passage of time (a few months). The images were 100x144 pixels each, except the test object of Q7, representing an eye region image of size 180x67 pixels. All test objects were displaying either full frontal or full profile views of the head, except in Q7. Except in Q8, in which the images contained 256 colors, all images lacked color but had 256 gray tones.

Q1 (Edges): The test object is a caricature image of a man, Figure 1. This image is obtained from a real photograph of the test object by a spatial high pass filtering. This filtering highlights certain edges by suppressing low spatial frequencies. Here it is achieved by taking the difference of the highest two resolutions (scales) of a Gaussian pyramid, obtained by using a true Gaussian filter with $\sigma = 1.22$, 2.

Q2 (Blur): The test object is a smoothed image of a woman, Figure 1. In contrast to question 1, the 10 test objects contain now only low frequencies. The low frequency images were obtained by using a Gaussian filter with $\sigma = 4.67$.

Q3 (Male-F-P), and Q4 (Female-F-P): The faces of the tests objects have frontal views whereas the faces of the object sets have profile views, Figure 2.

Q5 (MemM), Q6 (MemF): Face images are similar to Q3 and Q4 respectively except that the memorization task was added.

Q7 (Eyes): The test object consisted of the frontal eye region cut out of a photograph of an adult man. The picture did not show nose, mouth, or hair line. The object set displayed full frontal views of the face as in the object set of Q1. The question had the memorization task.

Q8 (Negatives): The test object was a color image of a female, whereas the images of the objects set were brightness and hue negated so that they looked like negative photographs, Figure 3 left. The negation was done in the HSV color space.

3 Experimental Results

Approximately 10'000 persons, with interest in science and technology in areas related to pattern recognition, were invited to do the test via www. As of the time of writing, 1 month after the invitation, 1838 persons, 492 women and 1346 men, took the test. Male dominance is a consequence of the male dominance in the call list. The fact that it is trivially simple to send false information on gender, age, ..etc has been handled as a statistical noise in our case for two reasons: i) we invited only a certain category of people, those who have related professional interests or activities to take the test, ii) nearly half of the female and male subjects did also send their names, addresses and emails. This enabled us to use the non-anonymous group as a control group for the anonymous group. Based on this we conclude that the noise from false data was negligible.

Given the number of categories (8 age categories, 8 continent categories, 2 gender categories), the number of participants, and the scope of this paper, we only report on the age category 21-30 and all continents. This age category contained 263 women and 680 men in our study. The statistics can be assumed to be representative for Caucasians since approximately 90 % of the participants were from Europe and USA. Although we report on one age category (the one on which we had most data), *also other age categories with significant number of subjects, confirm the conclusions reported below.*

The correct answers (CA) distribution represents the portion of the subjects that gave 0, 1, ...8 correct answers to the 8 questions across the male and the female subjects, Figure 1. The CA distribution evidences that the female subjects were more skilled to give the correct answers than the male subjects. The bias was observable even if the continent and the town type was restricted to be a small town (Halmstad, Sweden, 65'000 population) in Europe. The more correct answers, the less likely that these came from the male population e.g. it was approximately 2 times more likely that it was a woman who gave 6 correct answers than a man, 3 times more likely that it was a woman who gave 7 correct answers than a man...etc In other words the best female skills were far better than the best male skills although the female subject population had even in the average (median) more correct answers (5) than the male population (4).

Another novel result is that the high spatial frequencies carry more significant information for human face recognition than low spatial frequency information. This is evidenced by the fact that Q2 was much more difficult to answer correctly than Q1 (as well as all other questions), See Table 2. A totally random answer to Q2 would result in 10 % success which is suspiciously close to the female (16 %) and male (6 %) success rates. However the result of the subjects who have succeeded in this difficult question appears to be more due to skills than "luck". This is even more so for females than males because the females who had succeeded in Q2 were the top performers in all questions: they had in the average 6.2 (M: 5.4) correct answers with variance 1.5 (M: 2.5). Unskilled female and male subjects would have been represented equally well among those succeeded in Q2, had the results been due to random effects. In terms of the occurrence of

the next maximum in votes of Q2, both genders' agreed on the same answer: the correct answer. This is another support for skill as explanatory factor.

Our finding on the importance of high frequency data for human face recognition should be contrasted to the unimportance of these for some machine face recognition techniques based on small face pictures, i.e. lack of high frequencies, due to computational considerations.

For both women and men, the hair style was a significant distractor which is a reconfirmation of the results from previous studies, e.g. Bruce and Young, [3]. This was mostly evident from the answers to Q1, Q2, and Q8. The largest incorrect answers were always due to distractors had hairlines similar to hairlines of the test objects. This phenomenon was particularly striking in the CAs of Q2, in which 51 % of the female and 63 % of the male subjects were convinced that an erroneous answer was the correct answer. For both genders the largest (erroneous) answer coincided and the frequencies for the correct answers (F: 16 % M: 6 %) were the second largest. The hair style of the corresponding distractor that could cause so many subject's failure, was similar to that of the original test object, whereas the test object had changed her hair style completely in the image to be matched.

Q4 is a matching task between a frontal and a profile view of female objects, CA frequency were F: 70 % M: 47 % for male subjects. For both genders, Q4 was obviously more difficult to answer than Q3, having the CA frequencies F: 82 % M: 72 %, which concerns matching of a male object's frontal and profile views. The additional difficulty in Q4, resided in that there were two distraction elements: the hair style and the facial expression. From previous studies it is known that the recognition is hampered by a change in facial expression or hair style. But that the drop in performance is more significant for men than women has not been reported previously. By contrast, it is not easy to cross compare the CA's of Q3 and Q4 due to unequal difficulty level of the questions. In other words the numerical closeness of the CA's 72 % (female subjects on female objects, in Q4) and 70 % (male subjects on male objects, in Q3) concern questions of two different scales of difficulties.

With respect to Q5 and Q6, results show once more that the females performed significantly better. Again cross comparison of CA's between questions is more speculative than a comparison within the questions due to the different difficulty scales in the questions. For example one would be tempted to conclude that males recognizing females improved significantly in Q5 and Q6 as compared to questions Q3 and Q4. By contrast, the closeness of the CAs of males in Q5 and Q6 suggests that Q3, Q5 and Q6 were nearly equally difficult to answer. Independently, we find indeed that the female CA's for the same questions are nearly the same (82 %, 76 % and 82 %) too. This suggests that there is no additional bias of the memory on the skills of the genders, since the CA performance difference between the genders were nearly not altered by the memory capabilities.

The performance of the subjects in Q7 dropped indeed because the eye information is more restrictive, but the results are significantly above a 10% random selection, confirming that the females performed better than males.

Finally, concerning question 8, females performed again better than males. Apparently, negative images, although looking very different, still provide most if not all features necessary for recognition.

4 Conclusions

We have designed and implemented a human face recognition study that aimed at quantifying the skills of various categories of humans, including gender and age. The results uncovered that the female population have better skills in human face recognition tasks than the male population. We could reconfirm that the hair style and facial expressions are significant distraction factors for humans when they recognize faces. However, a novel finding is that men appear to have larger negative bias caused by these distractions than women.

We also found that the lack of details (high spatial frequencies) hampers the recognition significantly more than the lack of silhouette and shading (low spatial frequencies) information in general.

A more detailed quantification of our findings needs further studies.

Acknowledgment

We gratefully acknowledge the support of the Chalmers foundation and the use of M2VTS face database in our study, [10]. We thank Prof. J. M. H. du Buf, Univ. of Algarve, for his valuable comments on the manuscript.

References

1. A. J. Bruce and K. W. Beard. African Americans, and Caucasian Americans recognition and likability responses to African American and Caucasian American faces. *Journal of General Psychology*, 124:143–156, Apr 1997.
2. V. Bruce and A. Young. Understanding face recognition. *British Journal of Psychology*, 77:305–327, 1986.
3. V. Bruce and A. Young. *In the eye of the beholder*. Oxford University Press, Oxford, 1998.
4. P. Burt. Fast filter transforms for image processing. *Computer graphics and image processing*, 16:20–51, 1981.
5. H. D. Ellis, D. M. Ellis, and J. A. Hosie. Priming effects in childrens face recognition. *British Journal of Psychology*, 84:101–110, 1993.
6. H. W. Faw. Recognition of unfamiliar faces: Procedural and methodological considerations. *British Journal of Psychology*, 83:25–37, 1992.
7. L. Hassing, A. Wahling, and L. Backman. Minimal influence of age, education, and gender on episodic memory functioning in very old age: a population-based study of nonagenarians. *Archives of Gerontology and Geriatrics*, 27:75–87, 1998.
8. S. Pigeon and L. Vandendorpe. The M2VTS multi modal face database (release 1.0). In J. Bigun, G. Chollet, and G. Borgefors, editors, *Audio and Video based Person Authentication - AVBPA97*, pages 403–409. Springer, 1997.

Face Recognition by Auto-associative Radial Basis Function Network

Bai-ling Zhang ^{*} and Yan Guo

Kent Ridge Digital Labs (KRDL)
21 Heng Mui Keng Terrace, 119613, Singapore

Abstract. In this paper, we proposed an autoassociative Radial Basis Function (RBF) network and applied it with a modular structure to human face recognition. To capture the substantial facial features and reduce computational complexity, we propose to use wavelet transform (WT) to decompose face images and choose the lowest resolution sub-band coefficients for face representation. Results indicate that our scheme yields accurate recognition on the widely used XM2VTS face database and Olivetti Research Laboratory (ORL) face database.

1 Introduction

Though machine recognition of human faces has evolved through much progress, it remains a challenging task. In recent years, some biologically-motivated approaches seem to be promising in offering real solutions. The “eigenface” approach (Turk & Pentland 1991) is a well-known example, which represent faces by a linear combination of weighted eigenvectors, known as eigenfaces. However, there are several limitations accompanying with eigenface approach.

In the perceptual framework for human face processing (Hay and Young 1991), a concept of face recognition unit was suggested in which each unit produces a positive signal only for the particular person it is trained to recognise. In this framework, an adaptive learning model was proposed for face recognition (Howell and Buxton 1995), in which a personalised face recognition system was set up by exploiting RBF classifier with the “1-out-of-N encoding” principle.

Instead of setting up a classifier for each subject, we pursue another neural learning paradigm to represent faces, namely, auto-associative network model. In optical character recognition, auto-associative multilayer perceptron, known as autoencoder, has been successfully applied (Schwenk and Milgram 1995). The main idea is to use an autoassociative network with a low dimensional hidden layer for each class to recognise. Each network learns a hidden layer representation which preserves optimally the information of the examples of one class. These learned networks can be used like discriminant functions: the recognition error is in general much lower for examples of the learned class than for others.

It is well-known that wavelet based image representation has many advantages and there are strong evidences that human visual system processes the

^{*} email: bailing@krdl.org.sg

image in a multiscale way according to psychovisual research. Recently wavelet techniques have been applied to a range of problems in multimedia research, including face recognition. To capture the substantial facial features and reduce computational complexity, we propose to use wavelet transform (WT) to decompose face images and choose the lowest resolution subband coefficients for face representation.

2 Wavelet Transform

Wavelet transform (WT) decomposes an image into “subbands” that are localised in frequency and orientation. A wavelet transform is created by passing the image through a series of filter bank stages, in which the high-pass filter and low-pass filter are finite impulse response filters. The filtered outputs are then downsampled by a factor of 2 in the horizontal direction. These signals are then each filtered by an identical filter pair in the vertical direction. We end up with a decomposition of the image into 4 subbands denoted by LL, HL, LH, HH. Each of these subbands can be thought of as a smaller version of the image representing different image properties. The band LL is a coarser approximation to the original image. The bands LH and HL record the changes of the image along horizontal and vertical directions, respectively. The HH band shows the high frequency component of the image. Second level decomposition can then be conducted on the LL subband. Figure 1 shows a two-level wavelet decomposition of an image of size 200×150 pixels.

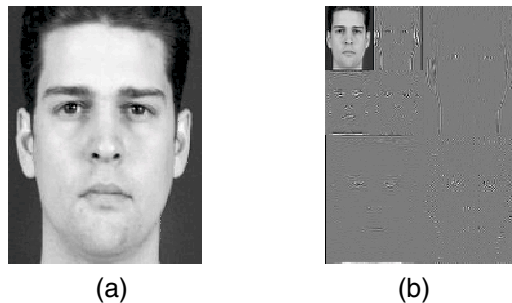


Fig. 1. (a). An original image with resolution 200×150 ; (b) the 2-level wavelet decomposition.

Earlier studies concluded that information in low spatial frequency bands play a dominant role in face recognition. Nastar et al (Nastar and Ayache 1996) have investigated the relationship between variations in facial appearance and their deformation spectrum. They found that facial expressions and small occlusion affect high-frequency spectrum only. Changes in pose or scale of a face affect

the intensity manifold globally, in which only their low-frequency spectrum is affected. Only a change in face will affect all frequency components. These phenomena have also been exploited in a recent work on combining wavelet subband representation with Eigenface method (Lai et al 2001). In the following, we'll use Daubechies D8 for image decomposition.

3 Auto-associative RBF Network as a Computation Model of Faces

We proposed an *Auto-associative Radial Basis Function (ARBF)* network, which is a 3-layer network with hidden layer performing a nonlinear mapping of the input layer onto the output layer. Its working involves first recording input patterns in terms of their distances from prototypes and then associating the distances with input vectors which are regarded as expected outputs. At the end of the recording phase, the input pattern are represented by M -dimensional vectors, where M is the number of hidden units. The output h for hidden unit k can be written as:

$$h_k = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_k\|^2}{2\sigma^2}\right) \quad (1)$$

which is an indication of the similarity between the input and the prototype. When an exemplar \mathbf{x} matches exactly the centre \mathbf{x}_k , the activity of the unit is at its maximum and it decreases as an exponential function of the squared distance between the input and the centre.

An output unit of ARBF network computes a weighted sum of hidden units outputs. The optimum values for the weights can be obtained by using a least-squares approximation. For M hidden unit activations $\mathbf{h}^{(n)} = [h_1^{(n)}, \dots, h_M^{(n)}]$, $n = 1, \dots, N$, using matrix representation $\mathbf{H} = [\mathbf{h}^{(1)}, \dots, \mathbf{h}^{(N)}]$, the connection weight \mathbf{W} from hidden layer to output can be calculated as

$$\mathbf{W} = \mathbf{XH}^+ \quad (2)$$

where \mathbf{H}^+ is the pseudo-inverse of \mathbf{H} .

Recall of the k -th face from the ARBF network is achieved by first input the face vector \mathbf{x}_k to the network and then premultiplying the hidden unit activation vector \mathbf{h}_k by the matrix \mathbf{W}

$$\hat{\mathbf{x}}_k = \mathbf{Wh}_k \quad (3)$$

where $\hat{\mathbf{x}}_k$ represents the estimation of the k -th face by the ARBF model. The quality of this estimation can be measured by computing the cosine of the angle between the vectors $\hat{\mathbf{x}}_k$ and \mathbf{x}_k , formally

$$\cos(\hat{\mathbf{x}}_k, \mathbf{x}_k) = \frac{\hat{\mathbf{x}}_k^T \mathbf{x}_k}{\|\hat{\mathbf{x}}_k\| \cdot \|\mathbf{x}_k\|} \quad (4)$$

with cosine of 1 indicating a perfect reconstruction of the stimulus.

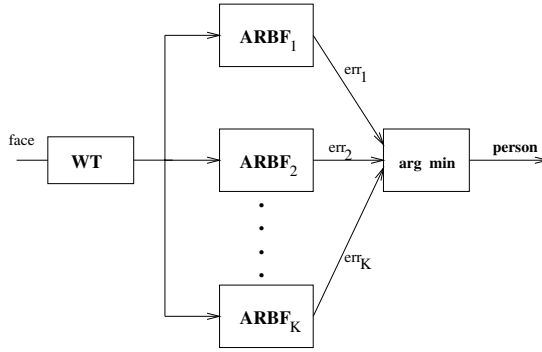


Fig. 2. The modular recognition scheme. In the model setting step, after decomposing a face image into wavelet subbands, the LL subband representation is used to construct a personalised Auto-associative Radial Basis Function (ARBF) model. In the recognition step, a probe face image is first decomposed by WT and the LL subband is inputted to all the ARBF models. The similarity scores are calculated and compared for all the estimations. A k th subject is identified as matching the probe if its ARBF gives the highest matching score.

4 Modular Face Recognition System

Our face recognition system consists of a set of subject-based separate ARBF modules, each capturing the variations of the respective subject. For a specific k th person, we first calculated an average face and a set of mean-centred vectors is obtained. After applying an L-level wavelet transform to decompose the reference images, a collection of LL subband image representations for each subject is used to construct an ARBF model according to Eq. (2).

When an unknown image \mathbf{x} is presented to the recognition stage, it is first subtracted by the average face and a caricature image is obtained. Then, an L-level WT is applied to transform the caricature image. The LL subband is represented as a probe image representation, which is applied to all ARBF models to yield respective estimations. Then, a similarity measurement between the probe image and an recalled image is performed to determine which recalled image representation best match the probe image representation, as given in Eq. 4. The process of identifying a face is demonstrated further in Figures 3 & 4.

5 Experimental Results

We conducted experiments on two different face databases. The first is XM2VTS face database from the University of Surrey, which consists of four images per person taken at a time interval of one month apart. The set of images is composed of frontal and near frontal images with varying facial expressions. The original image size is 726×576 pixels. In our experiments, the images were cropped and normalised to yield pixel size 150×200 . The second face database

we used is the Olivetti Research Laboratory (ORL) database, in which there are 40 persons and each person consists of 10 different facial views that represent various expressions, small occlusion (by glasses), different scale and orientations. The resolution of all images is 112×92 .

For the XM2VTS database, we selected three faces out of four for each subject to set up the respective ARBF model, and used the remaining face to test the recognition accuracy. After three level decomposition, the LL subband image has a size of 50×38 . The experiments repeated four times. For the ORL database, we randomly selected 3 faces out of 10 for each subject to set up an ARBF model and counted the recognition accuracy on the remaining faces. We applied two level wavelet decomposition, yielding 28×23 LL subband image representation. In Table 1, we illustrated the experiment results for the first face database by comparing different resolution levels of wavelet decomposition, which shows that three decomposition yields better recognition accuracy.

In the experiments, we also synthesised images by some simple geometric transformations, particularly rotation and scaling. We generated 10 synthetic images from each raw training image by making small, random perturbations to the original image: rotation (up to $+5$ and -5) and scaling (by a factor between 95% and 105 %)

For comparison reason, we downsampled face images from XM2VTS to 50×38 . The downsized face images are first used to set up personalised ARBF models and then the recognition is proceeded in the same way as described above. The averaged recognition accuracy is 78%, which showed that wavelet LL subband image representation do provide a better performance.

Table 1. Comparison of recognition accuracy for the XM2VTS face data exploiting different levels of wavelet decomposition.

level	2	3	4
Simulation I	77.29%	80.34%	78.31%
Simulation II	81.69%	85.42%	85.42%
Simulation III	82.03%	83.05%	81.69%
Simulation IV	83.73%	84.75%	83.39%
Average	81.19%	83.39%	82.20%

The recognition accuracy can be enhanced by rejecting some probe face images based on some thresholds. Denote the largest similarity score ρ_j and second largest score ρ_i . A face image is rejected from recognition if $\rho_j/\rho_i \leq \eta$. In Figure 4, we illustrate the accuracy versus the rejection rate which results from equally varying η from 0.01 to 0.1. From the simulation we see that the highest recognition accuracy is over 99.5% with rejection rate of 10% on testing set.

For the ORL face database, we compared our scheme with some methods reported in the literature (Sim et al 1999, Lawrence et al 1997, Turk & Pentland 1991). Table 2 summarises the results. The last row gives the recognition accuracy with our modular ARBF system. It is clear that the ARBF outperforms other methods.

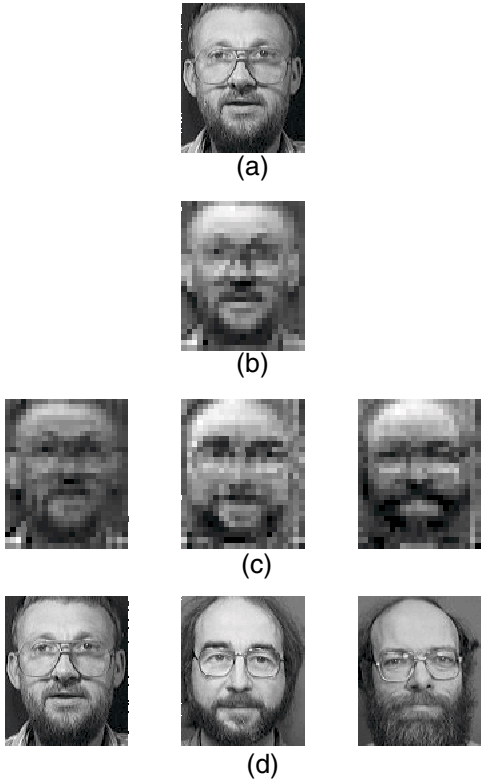


Fig. 3. Illustration of the face recognition process. (a). A probe image to be recognized; (b). Wavelet LL subband representation; (c). The first three recalled results from 40 ARBF models via the similarity measure Eq.(4); (d). The corresponding first three subjects. The most similar one (left) is recognized.

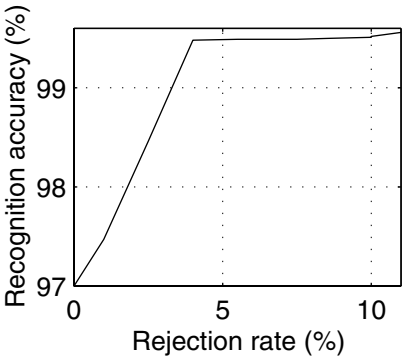


Fig. 4. Illustration of the recognition accuracy vs rejection rate for the ORL face database.

Table 2. Comparison of recognition accuracy for the ORL face data from different methods.

Images per person	3	5
Eigenface	81.8%	89.5%
SOM+CN	88.2%	96.5%
ARENA	92.2%	97.1%
ARBF	94.3%	98.2%

6 Discussionss

In this paper we proposed a modular face recognition scheme based on auto-associative RBF model, together with lower resolution wavelet subband coefficient for image representation. The efficiency of our scheme has been demonstrated for two standard databases, namely, XM2VTS face database from the University of Surrey, and faces database from Olivetti Research Laboratory. For the face database from XM2VTS, the averaged recognition accuracy is around 84%, while for the ORL database, the averaged recognition is over 98%, both without any rejection.

References

1. D.C.Hay, A. Young and A.W.Ellis, "Routes through the face recognition system", *Quarterly Journal of Experimental Psychology: Human Experimental Psychology*, Vol.43, pp. 761-791, 1991.
2. A.J. Howell and H. Buxton, "Receptive field functions for face recognition" University of Sussex Technical Reports, Dec. 1995.
3. C.Nastar, N.Ayache, "Frequency-based non-rigid motion analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, 18, pp.1067-1079, 1996.
4. J.-H.Lai, P.-C.Yuen, G.-C. Feng, "Face recognition using holistic Fourier invariant features", *Pattern Recognition*, 34, 95-109, 2001.
5. H.Schwenk and M.Milgram, "Transformation invariant autoassociation with application to handwritten character recognition", in *Neural Information Processing Systems (NIPS 7)*, D.S.Touretzyk, G.S.Tesauro and T.K.Leen eds., pp 991-998, MIT Press, 1995.
6. M.Turk and A.Pentland, "Eigenfaces for recognition", *Journal of Cognitive Neuroscience*, vol.3, pp.71-86, 1991.
7. S.Lawrence, C.L.Giles, A.C.Tsoi and A.D.Back, "Face recognition: A convolutional neural network approach," *IEEE Trans. Neural Network*, vol.8, pp.98-113, 1997.
8. T.Sim, R.Sukthankar, M.Mullin and S.Baluja, "High-performance memory-based face recognition for visitor identification," JPRC-TR-1999-001-1

Face Recognition Using Independent Component Analysis and Support Vector Machines *

O. Déniz **, M. Castrillón, and M. Hernández

Universidad de Las Palmas de Gran Canaria
Departamento de Informática y Sistemas
Edificio de Informática y Matemáticas
Campus Universitario de Tafra
35017 Las Palmas - Spain
{odeniz,mcastrillon,mhernandez}@dis.ulpgc.es

Abstract. Support Vector Machines (SVM) and Independent Component Analysis (ICA) are two powerful and relatively recent techniques. SVMs are classifiers which have demonstrated high generalization capabilities in many different tasks, including the object recognition problem. ICA is a feature extraction technique which can be considered a generalization of Principal Component Analysis (PCA). ICA has been mainly used on the problem of blind signal separation. In this paper we combine these two techniques for the face recognition problem. Experiments were made on two different face databases, achieving very high recognition rates. As the results using the combination PCA/SVM were not very far from those obtained with ICA/SVM, our experiments suggest that SVMs are relatively insensitive to the representation space. Thus as the training time for ICA is much larger than that of PCA, this result indicates that the best practical combination is PCA with SVM.

1 Introduction

The face recognition problem has attracted much research effort in the last years. Although it has proven to be a very difficult task even for frontal faces, certain algorithms can perform well under constrained conditions. The most prominent work was [1], which introduced the *eigenfaces* method, widely used as a reference. Recent advances in statistical learning theory, and in particular the introduction of Support Vector Machines [2] have made it possible to obtain very high accuracies for the object recognition problem. Independent Component Analysis [3] is also a relatively recent technique which has been mainly applied to *blind signal separation*, though it has been successfully applied to the face recognition problem too.

This paper is organized as follows. In the next two sections we give a brief introduction to ICA and SVM. In section 4 we describe the experiments carried out. Section 5 concludes by presenting future directions of research.

* Work partially funded by FEDER PI1999/153 research project.

** Author supported by graduate grant D260/54066308-R of Universidad de Las Palmas de Gran Canaria.

2 Independent Component Analysis

Independent Component Analysis is a technique for extracting statistically independent variables from a mixture of them [3]. ICA has been successfully applied to many different problems such as MEG and EEG data analysis, [4, 5, 6] finding hidden factors in financial data [7, 8] and face recognition (see [9] for an introduction and applications).

The ICA technique aims to find a linear transform for the input data using a basis as statistically independent as possible. Thus, ICA can be considered a generalization of Principal Component Analysis (PCA). PCA tries to obtain a representation of the inputs based on uncorrelated variables, whereas ICA provides a representation based on statistically independent variables. Figure 1 shows the difference between PCA and ICA basis images.



Fig. 1. Some original (left), PCA (center) and ICA (right) basis images for the Yale Face Database (see section 1).

In the context of face recognition, ICA has been showed to produce better results than those obtained with PCA, see [10] for a comparison. As opposed to PCA, ICA does not provide an intrinsic order for the representation coefficients of the face images. In [11] the best results were obtained with a order based on the ratio of between-class to within-class variance for each coefficient $r = \sigma_{between} / \sigma_{within}$, where $\sigma_{between} = \sum_j (\bar{x}_j - \bar{x})^2$ is the variance of the j class means and $\sigma_{within} = \sum_j \sum_i (x_{ij} - \bar{x})^2$ is the sum of the variances within each class.

3 Support Vector Machines

We only give here a brief presentation of the basic concepts needed. The reader is referred to [12] for a more detailed introduction and to [13] for a list of applications of SVMs. SVMs are based on structural risk minimization, which is the expectation of the test error for the trained machine. This risk is represented as $R(\alpha)$, α being the parameters of the trained machine. Let l be the number

of training patterns and $0 \leq \eta \leq 1$. Then, with probability $1 - \eta$ the following bound on the expected risk holds [2]:

$$R(\alpha) \leq R_{emp}(\alpha) + \sqrt{\frac{h(\log(2l/h) + 1) - \log(\eta/4)}{l}} \quad (1)$$

$R_{emp}(\alpha)$ being the empirical risk, which is the mean error on the training set, and h is the VC dimension. SVMs try to minimize the second term of (1), for a fixed empirical risk.

For the linearly separable case, SVM provides the optimal hyperplane that separates the training patterns. The optimal hyperplane maximizes the sum of the distances to the closest positive and negative training patterns. This sum is called *margin*. In order to weight the cost of missclassification an additional parameter is introduced. For the non-linear case, the training patterns are mapped onto a high-dimensional space using a kernel function. In this space the decision boundary is linear. The most commonly used kernel functions are polynomials, exponential and sigmoidal functions.

4 Experiments

In order to establish the performance of ICA/SVM, in comparison with other schemes, we carried out experiments on two independent face databases, the Yale Face Database [14], and a randomly chosen subset of the AR Face Database [15]. The Yale Face Database contains 165 images (11 per individual), with changes in facial expression, occlusion, and illumination conditions. From the AR Face Database we used 300 face images (12 per individual), with changes in facial expression and illumination conditions, and images taken in two sessions two weeks apart. All the results were obtained using 2-fold (AR) and 5-fold (Yale) cross-validation and varying the number of coefficients used in the range 1-N, N being the number of training images. ICA coefficients were ordered according to the ratio r explained in section 2. All the images were previously converted to 256 gray levels and histogram equalization was applied. The background in the Yale images was manually removed with a rectangle. For the images of the AR database a more plausible normalization was accomplished. Besides histogram equalization, we also performed geometric normalization. The images were firstly cropped with an ellipse, thus removing hair and shoulders. The eyes and mouth were located manually and then the images were shifted both in x and y and warped in order to have the eyes and mouth in the same place for all the images.

The ICA algorithm we used in our experiments was FastICA [16]. FastICA provides rapid convergence and estimates the independent components by maximizing a measure of independence among the estimated original components.

The SVM classifier, as introduced in section 3, is a 2-class classifier. Therefore, we had to adapt it to our multiclass problem. There are two options: using N SVMs (N being the number of classes), separating one class from the rest, or using $N(N - 1)/2$ SVMs, one for each pair of classes. As the accuracies between these two approaches are almost the same [17], we chose the first option, which

is less complex. The SVM algorithm used in the experiments presented problems of convergence when the input coefficients had a relatively high magnitude. This may be alleviated by dividing the input coefficients by a constant value (Thorsten Joachims, personal communication).

The results obtained appear in Table II. SVM was used only with polynomial (up to degree 3) and gaussian kernels (varying the kernel parameter σ).

Table 1. recognition rates obtained for Yale and AR images, using the Nearest Mean Classifier (NMC) and SVM. For SVM, a value of 1000 was used as missclassification weight. The last column represents the best results obtained varying σ . Note that the combination PCA-NMC corresponds to the *eigenfaces* method.

	NMC using euclidean distance	SVM			
		p=1	p=2	p=3	Gaussian
Yale PCA	92.73 %	98.79 %	98.79 %	98.79 %	99.39 %
ICA	95.76 %	99.39 %	99.39 %	99.39 %	99.39 %
AR PCA	48.33 %	92 %	91.67 %	91 %	92.67 %
ICA	70.33 %	93.33 %	93.33 %	92.67 %	94 %

For the Yale Database, there is no clear difference between ICA/SVM and PCA/SVM. We postulate that this is due to the fact that the classification error is too close to zero, which does not allow for differences to be seen clearly. As for the AR images, although the best absolute results are obtained with ICA and SVM, the performance is not far from that obtained with PCA and SVM. This is consistent with the results reported in [18], which suggested that SVM is relatively insensitive to the representation space.

Figure 2 represents the cumulative rank error and classification error as a function of the number of coefficients for the AR face set.

5 Conclusions and Future Work

We obtained experimental results showing that very high recognition rates can be achieved using ICA/SVM, although PCA/SVM also gave good results. Thus, evidence was given for the fact that SVMs are relatively insensitive to the representation space, which is in accordance with the results reported in [18], giving more importance to the trade-off between complexity and performance.

In future work we plan to use techniques such as SVM in the dynamic face recognition problem, the objective being the recognition of face sequences. The robustness of such system can be improved using temporal context.

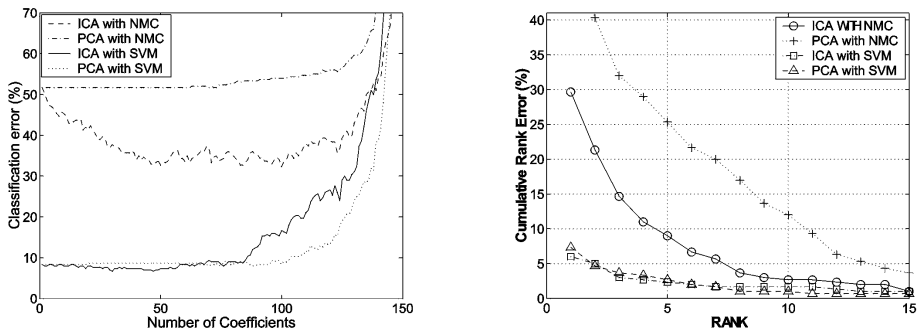


Fig. 2. Cumulative rank error and classification error as a function of the number of coefficients for the AR face set (SVM with the best absolute results obtained).

Acknowledgments

The authors would like to thank Dr. Marian Stewart Bartlett for her interesting comments on ICA. Thanks are also due to Prof. Dr. Robert Duin, David Tax, Thorsten Joachims and Prof. José Javier Lorenzo Navarro for valuable advice during the writing of this paper.

References

- [1] M. A. Turk and A.P. Pentland. Eigenfaces for Recognition. *Cognitive Neuroscience*, 3(1):71–86, 1991. <http://whitechapel.media.mit.edu/pub/images/>.
- [2] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, N.Y., 1995.
- [3] A. Bell and T. Sejnowski. An Information Maximization Approach to Blind Separation and Blind Deconvolution. *Neural Computation* 7, pages 1129–1159, 1995.
- [4] R. Vigário. Extraction of ocular artifacts from EEG using independent component analysis. *Electroenceph. clin. Neurophysiol.*, 103(3):395–404, 1997.
- [5] R. Vigário, V. Jousmäki, M. Hämmäläinen, R. Hari, and E. Oja. Independent component analysis for identification of artifacts in magnetoencephalographic recordings. In *Advances in Neural Information Processing Systems 10*, pages 229–235. MIT Press, 1998.
- [6] S. Makeig, A.J. Bell, T.P. Jung, and T.J. Sejnowski. Independent component analysis for electroencephalographic data. In *Advances in Neural Information Processing Systems 8*, pages 145–151. MIT Press, 1996.
- [7] K. Kiviluoto and E. Orja. Independent components analysis for parallel financial time series. In *Procs. ICONIP'98*, volume 2, pages 895–898, Tokyo, Japan, 1998.
- [8] A.D. Back and A.S. Weigend. A first application of independent component analysis to extracting structure from stock returns. *Int. J. on Neural Systems*, 4(8):473–484, 1998.
- [9] Aapo Hyvärinen and Erki Oja. Independent Component Analysis: A Tutorial. http://www.cis.hut.fi/~aapo/papers/IJCNN99_tutorialweb/, 1999.

- [10] Chengjun Liu and Harry Wechsler. Comparative Assessment of Independent Component Analysis (ICA) for Face Recognition. Second International Conference on Audio and Video-based Biometric Person Authentication, March 1999.
- [11] Marian Stewart Bartlett and Terrence J. Sejnowski. Independent Components of Face Images: a Representation for Face Recognition. In *Procs. of the 4th Annual Joint Symposium on Neural Computation*, Pasadena, CA, May 1997.
- [12] C.J.C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
- [13] SVM application list.
<http://www.clopinet.com/isabelle/Projects/SVM/applist.html>.
- [14] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.
- [15] A.M. Martinez and R. Benavente. The AR face database. Technical Report 24, CVC, June 1998.
- [16] The FastICA MATLAB package. Available at:
<http://www.cis.hut.fi/projects/ica/fastica/>
- [17] Olivier Chapelle, Patrick Haffner, and Vladimir Vapnik. SVMs for Histogram-based Image Classification. *IEEE Trans. on Neural Networks*, 10(5):1055–1065, September 1999.
- [18] K. T. Jonsson. *Robust Correlation and Support Vector Machines for Face Identification*. PhD thesis, School of Electronic Engineering, Information Technology and Mathematics, University of Surrey, UK, March 2000.

A Comparison of Face/Non-face Classifiers

Erik Hjelmås^{1,2} and Ivar Farup²

¹ Dept. of Informatics, Univ. of Oslo, P. O. Box 1080 Blindern, N-0316 Oslo, Norway

² Faculty of Technology, Gjøvik College, P. O. Box 191, N-2801 Gjøvik, Norway
{erikh,ivarf}@hig.no

Abstract. Most face detection algorithms can be divided into two sub-problems, initial visual guidance and face/non-face classification. In this paper we propose an evaluation protocol for face/non-face classification and provide experimental comparison of six algorithms. The overall best performing algorithms are the baseline template matching algorithms. Our results emphasize the importance of preprocessing.

1 Introduction

Face detection is an important and necessary first step in most face recognition applications. Face detection serves to localize potential face regions in images and classify them as faces or non-faces. This is a difficult task due to the dynamic appearance and variability of faces as opposed to more static objects such as vehicles or weapons. In addition to face recognition, areas such as content-based image retrieval, intelligent human-computer interfaces, crowd surveillance, video coding and email content security also make use of face detection algorithms.

The last decade has shown a great deal of research effort put into face detection technology. A comprehensive survey can be found in Hjelmås and Low [4], where the algorithms are classified as feature-based or image-based. However, not much work has been done on comparing existing algorithms. Some of the image-based algorithms report results on a common dataset (the CMU/MIT dataset), but there is not a common agreed upon evaluation protocol for this dataset. This has lead to different interpretations of testing parameters for this set, which makes it hard to compare the algorithms.

In this paper we provide an experimental comparison of six face detection algorithms, categorized as two baseline, two image-based and two feature-based algorithms. One of the feature-based algorithms is a new version of an existing technique, while the rest are implemented based on previously published papers by other authors. The algorithms are selected based on findings in [4], and also to represent significantly different approaches. We also propose an evaluation protocol for the face/non-face classifier in face detection algorithms.

In section 2, we present an overview of the dataset we have selected for training and testing, while section 3 describes the testing protocol in detail. Section 4 briefly presents the algorithms (since they are described in more detail elsewhere), section 5 contains the experimental results and discussion.

2 The Dataset

The dataset consists of images from the XM2VTS [7] and AR [6] face databases, and non-face images collected from the world wide web. The XM2VTS dataset is used for training. It contains 8 images of 295 subjects for a total 2360 images. All images are frontal view face images with a high degree of variation with respect to skin color, hair style, facial hair and glasses. The images are taken at four sessions with a month interval between sessions. For this training set, the coordinates of the eyes are available. For testing, we use the AR dataset with 3313 images from 136 subjects where most of the subjects images have been captured during two sessions with a 2 week interval between the sessions, from which we define the following subsets:

- Easy.** An easy dataset with 1783 face images. All subjects vary their facial expression, and there are large variations in lighting, but there are no facial occlusions. In 14% of the images the subjects were told to scream when the image was captured, thus these images have an extreme facial expression.
- Sunglasses.** A difficult dataset with 765 face images. All subjects are wearing dark sunglasses.
- Scarf.** A difficult dataset with 765 face images. All subjects are wearing a scarf covering the mouth area.

From the world wide web, we have collected manually a set of 67 large images with considerable structure, which might contain face-like patterns, which we use as the negative test set. In addition we have further collected a few large images for bootstrap training of the SNoW algorithm (described later).

The resolution of the training images (XM2VTS) are originally 720×576 , but we only use an extracted window covering the center of the face (rescaled to 20×20 or 60×60 pixels, and geometrically normalized with respect to the eyes). Similarly, the resolution of testing images (AR) are originally 768×576 , but we focus the search on subset covering the facial area (see the following section for details). The test sets and training sets are non-overlapping. All images are converted to 8 bit grayscale images (256 graylevels).

3 The Evaluation Protocol

Most face detection tasks can be divided into two steps, where the first step is an algorithm for visual guidance or simply an exhaustive search, and the second step is the actual face/non-face classification. In this section we propose a protocol for evaluating the second step. Not all proposed face detection algorithms work in this two-step fashion, but since the general problem of face detection can be decomposed into these two steps, all face detection approaches would benefit (in terms of accuracy) from decomposing or combining their algorithm this way, since the two steps does not depend on each other. Decomposing the problem leads to easier selection of the appropriate technique for the two sub-problems. The key elements of the evaluation protocol are the following (tailored to the datasets used in our experiments):

- The face classifiers generate a confidence score s_{algo} where $algo \in \{\mathbf{bE}, \mathbf{bC}, \mathbf{PCA}, \mathbf{SNoW}, \mathbf{Gradient}, \mathbf{Gabor}\}$ indicates the face classifier algorithm.
- The multiresolution scanning algorithm: a $n \times n$ window ω scans the entire image with 1 pixel step size and the image is subsampled by a factor of 1.2 until all scales and locations have been included. The face classifiers are applied at each location and scale. n is set to 20 for the image-based and baseline algorithms, and 60 for the feature-based algorithms. However, the 20×20 windows are just downsampled versions of the 60×60 windows in order to have the same number of testing windows ω for all algorithms.
- A correct detection of a face in a face image I is registered if the window ω which produces the highest confidence score ($\max_{\omega}(s_{algo})$) is *correctly centered* in I . We have manually located the center (x_c, y_c) of the face for all the test images, so we define ω correctly centered to be ω located such that its center region $\{(\frac{n}{2} \pm \frac{n}{4}, \frac{n}{2} \pm \frac{n}{4})\}$ encompasses (x_c, y_c) .
- The correct face detection rate CD is simply

$$CD_{testset} = \frac{\text{number of images with face correctly detected}}{\text{total number of images}}$$

where $testset \in \{\mathbf{Easy}, \mathbf{Sunglasses}, \mathbf{Scarf}\}$ indicates the test set used, and the total number of images is 1783 for the **Easy** dataset and 765 for the **Scarf** and **Sunglasses** dataset. We know that for the face images there is only one face present in each image.

- For the false alarm rate FA , we are simply interested in the number of false alarms relative to the total number of windows ω produced by the multiresolution scanning algorithm on the negative test set. This number is 5938360, so the false alarm rate is computed from

$$FA = \frac{\text{number of false detections}}{5938360}$$

A false alarm is a window ω where the face classifier produces a $s_{algo} > t_{algo}$. We do not count false alarms in the face images (the positive test sets).

- Results are reported in terms of ROC (Receiver Operator Characteristics) curves, which shows the trade-off between correct face detection rate CD and the false alarm rate FA . The threshold t_{algo} for the face classifier is varied in a range to produce a false alarm rate $10^{-4} \leq FA \leq 10^{-1}$.

4 The Algorithms

Baseline Template Algorithms. Standard template matching is used as baseline algorithms for comparison. The training images are geometrically normalized such that a 20×20 window encompassing eyes in fixed positions, nose and mouth, can be extracted. We compute the template by simply averaging these training images. Matching is performed by measuring either Euclidean distance (**bE**) or computing the normalized correlation coefficient (**bC**) between template and testing window.

Image-Based: PCA Algorithm. Principal Component Analysis (PCA) can be used to create a face space consisting of eigenfaces as an orthogonal basis [11], on which new faces can be projected to achieve a more compact representation. In our implementation, we use the reconstruction error $\epsilon^2 = \|\tilde{\omega}\| - \sum_{i=1}^n y_i^2$ (where y_i are projection coefficients and $\|\tilde{\omega}\|$ is the mean subtracted window) as a measure for the score s_{pca} . We only keep the first principal component for representation (thus $n = 1$).

Image-Based: SNoW Algorithm. The SNoW (Sparse Network of Winnows) learning architecture, proposed by Roth in [1], has been successfully applied to face detection by Roth et al. in [10]. We have implemented this algorithm using the software available from the website of Roth for training, and our own implementation for testing. The technical details of the algorithm are described in [10]. We also use a training procedure similar to the bootstrap training proposed by Sung and Poggio [13].

Feature-Based: Gradient Algorithm. This algorithm is a variant of the work of Maio and Maltoni [3]. From a 60×60 window a directional image consisting of 20×20 pairs of directions and weights is extracted using the algorithm by Donahue and Rokhlin [4]. This is compared with a constructed template representing the characteristic features of a face using the distance function from [6]. In contrast with the original work of Maio and Maltoni, the constructed template does not contain the ellipsis outlining the face, and the distances between the facial elements in the constructed template are chosen to resemble the template used in the baseline algorithms as closely as possible.

Feature-Based: Gabor Algorithm. Gabor features are widely applied for local feature extraction in face recognition systems and have also been used for face detection [9] and facial feature detection [12]. Gabor features are extracted using a set of 2D Gabor filters [2]. In our implementation we use a set of 40 filters (5 sizes, 8 orientations) generated by a wavelet expansion. We create a Gabor template which is a 60×60 window where the set of 40 Gabor coefficients have been extracted at the two locations corresponding to the eyes. In other words, we have a template which simply represents the average eyes. We only keep the magnitude of the complex coefficients and compare the template with the extracted subwindow at each location using the normalized correlation coefficient.

5 Results and Discussion

We try out several combinations of two preprocessing techniques – subtraction of best fit linear plane and histogram equalization – using the **bE**-algorithm. Figure 1A shows that the preprocessing is of major importance for the algorithm to work correctly. The best *CD* is obtained when both kinds of preprocessing are applied to both the test images and the template. This combination is thus applied for the remaining algorithms (except for the Gabor algorithm which is not as dependent on preprocessing).

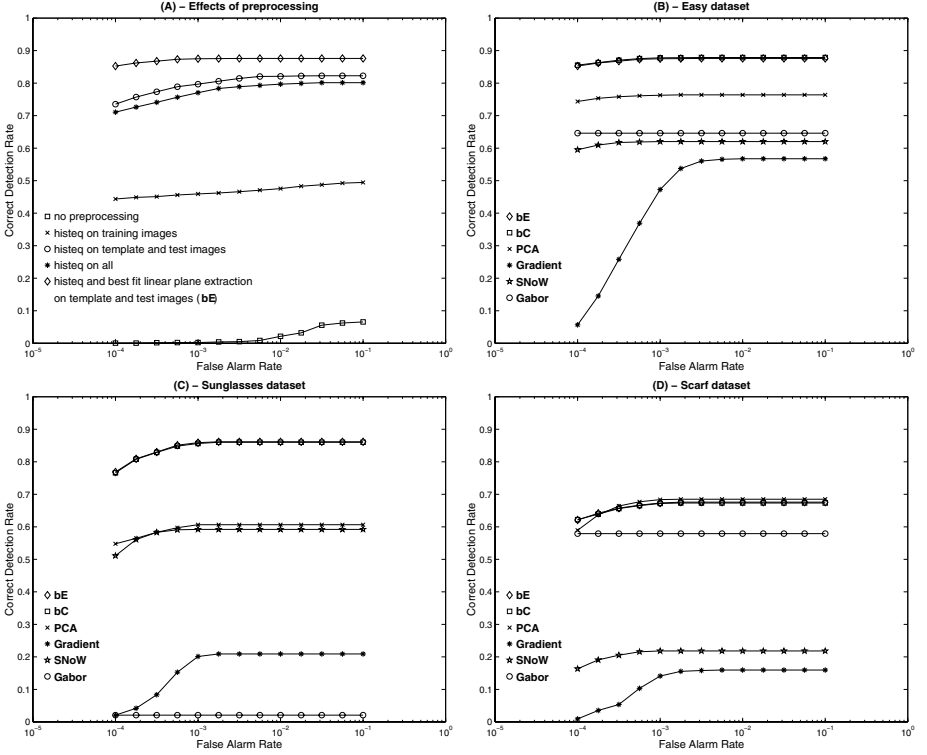


Fig. 1. Experimental results.

The results for all algorithms are shown in figure 1B for the **easy** dataset, figure 1C for the **sunglasses** dataset, and figure 1D for the **scarf** dataset. The baseline template matching algorithms are the overall best performing algorithms.

The **PCA** algorithm gives the best results when using only the first principal component, thus reducing the algorithm to a modified correlation measure. The reason for this is possibly that the size of the training set is not large enough to provide a general basis for representing the class of faces. We believe that this could be the reason since a general face class consisting of geometrically normalized faces should be Gaussian [8], and examination of the training data when plotting the projection coefficients of the first two principal components showed us that this is not the case.

The size of the training set is possibly also the reason to the poor performance of the **SNoW** classifier, since the classifier had no problems learning the face/non-face classification during training and initial testing.

The abandoning of the ellipsis around the face introduced an important alteration for the **Gradient** algorithm compared to the original work of Maio and Maltoni [6]. This might explain why the algorithm performs less than ideally.

In the original work, the total weight of the ellipsis in the distance function was approximately 2–3 times the weight of the remaining template, indicating the importance of the ellipses.

Selection of the Gabor filters for the **Gabor** algorithm was accomplished by manual inspection, and we have no reason to believe that these filters are optimal for representing the face class (in terms of the eyes here).

To our knowledge, detailed comparison of the preprocessing effects in face detection has not been presented earlier, thus figure 10A is quite significant. Simple template matching algorithms are not always used as a baseline for comparison, and our results should be taken as a strong indication that this is necessary. Due to the complexity of the other algorithms such as different selection of training set size, training parameters, template and filter design, (and the fact that we are mostly using reimplementations of the algorithms) improved performance can most likely be achieved. However, in our scenario, the simple baseline algorithms show impressive performance with the right kind of preprocessing.

References

- [1] A. J. Carlson, C. M. Cumby, J. L. Rosen, and D. Roth. SNoW user's guide. Technical Report UIUC-DCS-R-99-210, UIUC CS dept, 1999.
- [2] J. G. Daugman. Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Research*, 20:847–856, 1980.
- [3] M. J. Donahue and S. I. Rokhlin. On the use of level curves in image analysis. *Image Understanding*, 57:185–203, 1993.
- [4] E. Hjelmås and B. K. Low. Face detection: A survey. submitted.
- [5] D. Maio and D. Maltoni. Real-time face location on gray-scale static images. *Pattern Recognition*, 33:1525–1539, 2000.
- [6] A. M. Martinez and R. Benavente. The AR face database. Technical Report CVC 24, School of Elec. and Comp. Eng., Purdue University, 1998.
- [7] K. Messer, J. Matas, J. Kittler, J. Luetttin, and G. Maitre. XM2VTSDB: The extended M2VTS database. In *Second International Conference on Audio and Video-based Biometric Person Authentication*, 1999.
- [8] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(1), 1997.
- [9] K. Okada, J. Steffens, T. Maurer, H. Hong, E. Elagin, H. Neven, and C. v. d. Malsburg. The Bochum/USC face recognition system and how it fared in the FERET phase III test. In *Face Recognition: From Theory to Application*. Springer, 1998.
- [10] D. Roth, M.-H. Yang, and N. Ahuja. A SNoW-based face detector. In *Advances in Neural Information Processing Systems 12 (NIPS 12)*. MIT press, 2000.
- [11] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America*, 4:519–524, 1987.
- [12] F. Smeraldi, O. Carmona, and J. Bigün. Saccadic search with Gabor features applied to eye detection and real-time head tracking. *Image and Vision Computing*, 18:323–329, 2000.
- [13] K.-K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, January 1998.

Using Mixture Covariance Matrices to Improve Face and Facial Expression Recognitions

Carlos E. Thomaz¹, Duncan F. Gillies¹, and Raul Q. Feitosa²

¹ Imperial College of Science Technology and Medicine, Department of Computing
180 Queen's Gate, London SW7 2BZ, United Kingdom
{cet,dfg}@doc.ic.ac.uk

² Catholic University of Rio de Janeiro, Department of Electrical Engineering
r. Marques de Sao Vicente 225, Rio de Janeiro 22453-900, Brazil
raul@ele.puc-rio.br

Abstract. In several pattern recognition problems, particularly in image recognition ones, there are often a large number of features available, but the number of training samples for each pattern is significantly less than the dimension of the feature space. This statement implies that the sample group covariance matrices often used in the Gaussian maximum probability classifier are singular. A common solution to this problem is to assume that all groups have equal covariance matrices and to use as their estimates the pooled covariance matrix calculated from the whole training set. This paper uses an alternative estimate for the sample group covariance matrices, here called the mixture covariance, given by an appropriate linear combination of the sample group and pooled covariance matrices. Experiments were carried out to evaluate the performance associated with this estimate in two recognition applications: face and facial expression. The average recognition rates obtained by using the mixture covariance matrices were higher than the usual estimates.

1 Introduction

A critical issue for the Gaussian maximum probability classifier is the inverse of the sample group covariance matrices. Since in practice these matrices are not known, estimates must be computed based on the observations (patterns) available in a training set. In some applications, however, there are often a large number of features available, but the number of training samples for each group is limited and significantly less than the dimension of the feature space. This implies that the sample group covariance matrices will be singular.

This problem, which is called a small sample size problem [5], is quite common in pattern recognition, particularly in image recognition where the number of features is very large. One way to overcome this problem is to assume that all groups have equal covariance matrices and to use as their estimates the weighting average of each sample group covariance matrix, given by the pooled covariance matrix calculated from the whole training set.

This paper uses another estimate for the sample group covariance matrices [4], here called mixture covariance matrices, given by an appropriate linear combination of the sample group covariance matrix and the pooled covariance one. The mixture covariance matrices have the property of having the same rank as the pooled estimate, while allowing a different estimate for each group. Thus, the mixture estimate may result in higher accuracy.

In order to evaluate this approach, two pattern recognition applications were considered: face recognition and facial expression recognition. The evaluation used different image databases for each application. A probabilistic model was used to combine the well-known

dimensionality reduction technique called Principal Component Analysis (PCA) and the Gaussian maximum probability classifier, and in this way we could investigate the performance of the mixture covariance matrices on the referred recognition tasks. Experiments carried out show that the mixture covariance estimates attained the best performance in both applications.

2 Dimensionality Reduction

One of the most successful approaches to the problem of creating a low dimensional image representation is based on Principal Component Analysis (PCA). PCA generates a set of orthonormal basis vectors, known as principal components, that minimizes the mean square reconstruction error and describe major variations in the whole training set considered.

Instead of analyzing the maximum probability classifier directly on the face or facial expression images, PCA is applied first, to provide dimensionality reduction. As the number of training samples is limited and significantly less than the number of pixels of each image, the high-dimensional space is very sparsely represented, making the parameter estimation quite difficult a problem that is called the curse of dimensionality [8]. Furthermore, many researchers have confirmed that the PCA representation has good generalization ability especially when the distributions of each class are separated by the mean difference [1,6,7,9].

3 Maximum Probability Classifier

The basic problem in the decision-theoretic methods for pattern recognition consists of finding a set of g discriminant functions $d_1(\mathbf{x})$, $d_2(\mathbf{x})$, ..., $d_g(\mathbf{x})$, where g is the number of groups or classes, with the decision rule such that if the p -dimensional pattern vector \mathbf{x} belongs to the class i ($1 \leq i \leq g$), then $d_i(\mathbf{x}) \geq d_j(\mathbf{x})$, for all $i \neq j$ and $1 \leq j \leq g$.

The Bayes classifier designed to maximize the total probability of correct classification, where equal prior probabilities for all groups are assumed, corresponds to a set of discriminant functions equal to the corresponding probability density functions, that is, $d_i(\mathbf{x}) = f_i(\mathbf{x})$ for all classes [8]. The most common probability density function applied to pattern recognition systems is based on the Gaussian multivariate distribution

$$d_i(x) = f_i(x | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) = \frac{1}{(2\pi)^{p/2} |\boldsymbol{\Sigma}_i|^{1/2}} \exp \left[-\frac{1}{2} (x - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (x - \boldsymbol{\mu}_i) \right], \quad (1)$$

where $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ are the class i population mean vector and covariance matrix. Usually the true values of the mean and the covariance matrix are seldom known and must be estimated from training samples. The mean is estimated by the usual sample mean

$$\boldsymbol{\mu}_i \equiv \bar{x}_i = \frac{1}{k_i} \sum_{j=1}^{k_i} x_{i,j}, \quad (2)$$

where $x_{i,j}$ is observation j from class i , and k_i is the number of training observations from class i . The covariance matrix is commonly estimated by the sample group covariance matrix defined as

$$\Sigma_i \equiv S_i = \frac{1}{(k_i - 1)} \sum_{j=1}^{k_i} (x_{i,j} - \bar{x}_i)(x_{i,j} - \bar{x}_i)^T. \quad (3)$$

From replacing the true values of the mean and the covariance matrix in (1) by their respective estimates, the Bayes decision rule achieves optimal classification accuracy only when the number of training samples increases toward infinity [4]. In fact for p -dimensional patterns the sample covariance matrix is singular if less than $p + 1$ training samples from each class i are available, that is, the sample covariance matrix can not be calculated if k_i is less than the dimension of the feature space.

One method routinely applied to solve this problem is to assume that all classes have equal covariance matrices, and to use as their estimates the pooled covariance matrix. This covariance matrix is a weighting average of each sample group covariance matrix and, assuming that all classes have the same number of training observations, is given by

$$S_{pooled} = \frac{1}{g} \sum_{i=1}^g S_i. \quad (4)$$

Since more observations are taken to calculate the pooled covariance matrix S_{pooled} , this one will potentially have a higher rank than S_i and will be eventually full rank. Although the pooled estimate does provide a solution for the algebraic problem arising from the insufficient number of training samples in each group, assuming equal covariance for all groups may bring about distortions in the modeling of the classification problem and consequently lower accuracy.

4 Mixture Covariance Matrix

The choice between the sample group covariance matrix and the pooled covariance one represents a restrictive set of estimates for the true covariance matrix. A less limited set can be obtained using the mixture covariance matrix.

4.1 Definition

The mixture covariance matrix is a linear combination between the pooled covariance matrix S_{pooled} and the sample covariance matrix of each class S_i . It is given by

$$Smix_i(w_i) = w_i S_{pooled} + (1 - w_i) S_i. \quad (5)$$

The mixture parameter w_i takes on values $0 < w_i \leq 1$ and is different for each class. This parameter controls the degree of shrinkage of the sample group covariance matrix estimates toward the pooled one.

Each $Smix_i$ matrix has the important property of admitting an inverse if the pooled estimate S_{pooled} does so [2]. This implies that if the pooled estimate is non-singular and the mixture parameter takes on values $w_i > 0$, then the $Smix_i$ will be non-singular.

Then the remaining question is: what is the value of the w_i that gives a relevant linear mixture between the pooled and sample covariance estimates? A method that determines an appropriate value of the mixture parameter is described in the next section.

4.2 The Mixture Parameter

According to Hoffbeck and Landgrebe [4], the value of the mixture parameter w_i can be appropriately selected so that a best fit to the training samples is achieved. Their technique is based on the leave-one-out-likelihood (L) parameter estimation.

In the L method, one sample of the class i training set is removed and the mean and covariance matrix from the remaining $k_i - 1$ samples are estimated. Then the likelihood of the excluded sample is calculated given the previous mean and covariance matrix estimates. This operation is repeated $k_i - 1$ times and the average log likelihood is computed over all the k_i samples. Their strategy is to evaluate several different values of w_i in the range $0 < w_i \leq 1$, and then choose w_i that maximizes the average log likelihood. Once the mixture parameter w_i is selected, the proposed covariance matrix estimate is calculated using all the training samples and replaced into the maximum probability classifier defined in (1).

The mean of class i without sample r may be computed as

$$\bar{x}_{i \setminus r} = \frac{1}{(k_i - 1)} \left[\sum_{j=1}^{k_i} x_{i,j} - x_{i,r} \right]. \quad (6)$$

The notation $\setminus r$ indicates the corresponding quantity is calculated with the r^{th} observation from class i removed. Following the same idea, the sample covariance matrix and the pooled covariance matrix of class i without sample r are

$$S_{i \setminus r} = \frac{1}{(k_i - 2)} \left[\sum_{j=1}^{k_i} (x_{i,j} - \bar{x}_{i \setminus r})(x_{i,j} - \bar{x}_{i \setminus r})^T - (x_{i,r} - \bar{x}_{i \setminus r})(x_{i,r} - \bar{x}_{i \setminus r})^T \right], \quad (7)$$

$$S_{\text{pooled}_{i \setminus r}} = \frac{1}{g} \left[\sum_{j=1}^g S_j - S_i + S_{i \setminus r} \right]. \quad (8)$$

Then the average log likelihood of the excluded samples can be calculated as follows:

$$\bar{L}_i(w_i) = \frac{1}{k_i} \left[\sum_{r=1}^{k_i} \ln[f(x_{i,r} | \bar{x}_{i \setminus r}, \text{Smix}_{i \setminus r}(w_i))] \right], \quad (9)$$

where $f(x_{i,r} | \bar{x}_{i \setminus r}, \text{Smix}_{i \setminus r}(w_i))$ is the Gaussian probability function defined in (1) with $\bar{x}_{i \setminus r}$ mean vector and $\text{Smix}_{i \setminus r}(w_i)$ covariance matrix defined as

$$\text{Smix}_{i \setminus r}(w_i) = w_i S_{\text{pooled}_{i \setminus r}} + (1 - w_i) S_{i \setminus r}. \quad (10)$$

This approach, if implemented in a straightforward way, would require computing the inverse and determinant of the $\text{Smix}_{i \setminus r}(w_i)$ for each training sample. As the $\text{Smix}_{i \setminus r}(w_i)$ is a p by p matrix and p is typically a large number, this computation would be quite expensive. Hoffbeck and Landgrebe [4], using the Sherman-Morrison-Woodbury formula [3], have showed that it is possible to significantly reduce the required computation by writing the log likelihood of the excluded samples in a form as follows:

$$\ln[f(x_{i,r} | \bar{x}_{i \setminus r}, \text{Smix}_{i \setminus r}(w_i))] = -\frac{1}{2} \ln[|Q|(1 - vd)] - \frac{1}{2} \left(\frac{k_i}{k_i - 1} \right) \left[\frac{d}{1 - vd} \right], \quad (11)$$

where

$$Q = \left[(1 - w_i) \frac{(k_i - 1)}{(k_i - 2)} + w_i \frac{1}{g(k_i - 2)} \right] S_i + w_i S_{pooled}, \quad (12)$$

$$v = \frac{k_i}{(k_i - 1)(k_i - 2)} \left[1 - w_i \frac{(g - 1)}{g} \right], \quad (13)$$

$$d = (x_{i,r} - \bar{x}_i)^T Q^{-1} (x_{i,r} - \bar{x}_i). \quad (14)$$

5 Experiments

Two experiments with two different databases were performed.

In the face recognition experiment the ORL Face Database containing ten images for each of 40 individuals, a total of 400 images, were used. The Tohoku University has provided the database for the facial expression experiment. This database is composed of 193 images of expressions posed by nine Japanese females. Each person posed three or four examples of each six fundamental facial expression: anger, disgust, fear, happiness, sadness and surprise. The database has at least 29 images for each fundamental facial expression. For implementation convenience all images were first resized to 64x64 pixels.

The experiments were carried out as follows. First PCA reduces the dimensionality of the original images and secondly the Gaussian maximum probability classifier using one out of the three covariance estimates (S_i , S_{pooled} and S_{mix_i}) was applied. Each experiment was repeated 25 times using several PCA dimensions. Distinct training and testing sets were randomly drawn, and the mean and standard deviation of the recognition rate were calculated.

The face recognition classification was computed using for each individual 5 images to train and 5 images to test. In the facial expression recognition, the training and test sets were respectively composed of 20 and 9 images. The size of the mixture parameter ($0 < w_i \leq 1$) optimization range was taken to be 20, that is $w_i = [0.05, 0.10, 0.15, \dots, 1]$.

6 Results

Tables 1 and 2 present the training and test average recognition rates (with standard deviations) of the face and facial expression databases, respectively, over the different PCA dimensions.

Since only 5 images of each individual were used to form the face recognition training set, the results relative to the sample group covariance estimate were limited to 4 PCA components. Table 1 shows that in all but one experiment the S_{mix} estimate led to higher accuracy than did both the pooled covariance and sample group covariance matrices. In terms of how sensitive the mixture covariance results were to the choice of the training and test sets, it is fair to say that the S_{mix} standard deviations were similar to the other two covariance estimates.

Table 2 shows the results of the facial expression recognition. For more than 20 components when the sample group covariance estimate became singular, the mixture covariance estimate reached higher recognition rates than the pooled covariance estimate. Again, regarding the computed standard deviations, the *Smix* estimate showed to be as sensitive to the choice of the training and test sets as the other two estimates.

Table 1. Face Recognition Results.

PCA	Sgroup		Spooled		Smix	
Components	Training	Test	Training	Test	Training	Test
4	99.5 (0.4)	51.6 (4.4)	73.3 (3.1)	59.5 (3.0)	90.1 (2.1)	70.8 (3.2)
10			96.6 (1.2)	88.4 (1.4)	99.4 (0.5)	92.0 (1.5)
20			99.2 (0.6)	91.8 (1.8)	100.0 (0.1)	94.5 (1.7)
30			99.9 (0.2)	94.7 (1.7)	100.0 (0.0)	95.9 (1.5)
40			100.0 (0.0)	95.4 (1.5)	100.0 (0.0)	96.2 (1.6)
50			100.0 (0.0)	95.7 (1.2)	100.0 (0.0)	96.4 (1.5)
60			100.0 (0.0)	95.0 (1.6)	100.0 (0.0)	95.8 (1.6)
70			100.0 (0.0)	94.9 (1.6)	100.0 (0.0)	95.4 (1.6)

Table 2. Facial Expression Recognition Results.

PCA	Sgroup		Spooled		Smix	
Components	Training	Test	Training	Test	Training	Test
5	41.5 (4.2)	20.6 (3.9)	32.3 (3.0)	21.6 (3.8)	34.9 (3.3)	21.3 (4.1)
10	76.3 (3.6)	38.8 (5.6)	49.6 (3.9)	26.5 (6.8)	58.5 (3.7)	27.9 (5.6)
15	99.7 (0.5)	64.3 (6.4)	69.1 (3.6)	44.4 (5.3)	82.9 (2.9)	49.7 (7.7)
20			81.2 (2.6)	55.9 (7.7)	91.4 (2.8)	61.3 (7.1)
25			86.9 (2.8)	64.9 (6.9)	94.8 (2.2)	68.3 (5.1)
30			91.9 (1.7)	70.1 (7.8)	96.8 (1.3)	72.3 (6.2)
35			94.3 (1.7)	72.0 (7.4)	97.7 (1.1)	75.6 (5.5)
40			95.9 (1.4)	75.6 (7.1)	98.3 (1.1)	77.2 (5.7)
45			96.7 (1.3)	78.4 (6.5)	98.6 (0.8)	79.1 (5.4)
50			97.6 (1.0)	79.4 (5.8)	99.2 (0.7)	81.0 (6.6)
55			98.5 (0.9)	81.6 (6.6)	99.5 (0.6)	82.8 (6.3)
60			99.1 (0.8)	82.1 (5.9)	99.6 (0.6)	83.6 (7.2)
65			99.5 (0.6)	83.3 (5.5)	99.8 (0.4)	84.5 (6.2)

7 Conclusion

This paper used an estimate for the sample group covariance matrices, here called mixture covariance matrices, given by an appropriate linear combination of the sample group covariance matrix and the pooled covariance one. The mixture covariance matrices have the property of having the same rank as the pooled estimate, while allowing a different estimate for each group.

Extensive experiments were carried out to evaluate this approach on two recognition tasks: face recognition and facial expression recognition. A Gaussian maximum probability

classifier was built using the mixture estimate and the typical sample group and pooled estimates. In both tasks the mixture covariance estimate achieved the highest accuracy. Regarding the sensitiveness to the choice of the training and test sets, the mixture covariance matrices presented similar performance to the other two usual estimates.

Acknowledgments

The first author was partially supported by the Brazilian Government Agency CAPES.

References

1. C. Liu and H. Wechsler, Learning the Face Space Representation and Recognition . In Proc. 15th Int'l Conference on Pattern Recognition, ICPR 2000, Barcelona, Spain, September 2000.
2. C.E. Thomaz, R.Q. Feitosa, and A. Veiga, Separate-Group Covariance Estimation with Insufficient Data for Object Recognition . In Proc. Fifth All-Ukrainian International Conference, pp. 21-24, Ukraine, November 2000.
3. G.H. Golub and C.F. Van Loan, *Matrix Computations*, second edition. Baltimore: Johns Hopkins Univ. Press, 1989.
4. J.P. Hoffbeck and D.A. Landgrebe, "Covariance Matrix Estimation and Classification with Limited Training Data", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 18, no. 7, July 1996.
5. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, second edition. Boston: Academic Press, 1990.
6. M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 12, No. 1, Jan. 1990.
7. M. Turk and A. Pentland, Eigenfaces for Recognition, Journal of Cognitive Neuroscience, Vol. 3, pp. 72-85, 1991.
8. R.A. Johnson R.A. and D.W. Wichern, *Applied Multivariate Statistical Analysis*, by Prentice-Hall, Inc., 3d. edition, 1992.
9. W. Zhao, R. Chellappa and A. Krishnaswamy, Discriminant Analysis of Principal Components for Face Recognition, Proc. 2nd International Conference on Automatic Face and Gesture Recognition, pp. 336-341, Japan, April, 1998.

Real-Time Face Detection Using Edge-Orientation Matching

Bernhard Fröba and Christian Külbeck

Fraunhofer-Institute for Integrated Circuits
Am Weichselgarten 3, D-91058 Erlangen, Germany
{bdf,kue}@iis.fhg.de

Abstract. In this paper we describe our ongoing work on real-time face detection in grey level images using edge orientation information. We will show that edge orientation is a powerful local image feature to model objects like faces for detection purposes. We will present a simple and efficient method for template matching and object modeling based solely on edge orientation information. We also show how to obtain an optimal face model in the edge orientation domain from a set of training images. Unlike many approaches that model the grey level appearance of the face our approach is computationally very fast. It takes less than 0.08 seconds on a Pentium II 500MHz for a 320x240 image to be processed using a multi-resolution search with six resolution levels. We demonstrate the capability of our detection method on an image database of 17000 images taken from more than 2900 different people. The variations in head size, lighting and background are considerable. The obtained detection rate is more than 93% on that database.

1 Introduction

A number of current and probably many future applications such as complex interaction in multi-modal human machine interfaces, face recognition, video coding and virtual reality applications require the fast and reliable visual detection of faces. The algorithms therefore should be robust under a wide range of acquisition conditions such as lighting and background variations. In the past, several approaches for face detection have been made. Most of the fast algorithms use color information for the segmentation of skin-tone-like areas. These areas are usually clustered and searched for facial features. See [13, 12, 12, 18] for reference. Another widely-used class of methods for finding faces uses various kinds of grey level correlation approaches. The majority of the approaches [10, 5, 10, 15] use a separate class for each the faces and non-faces to model the problem domain. Most of these methods are reported to have a good detection performance but the drawback is to be computationally rather expensive and are at present not the appropriate choice for use in real-time applications. Since we wanted to have a method which works on video streams as well as still grey images, we could not exploit motion or color information. Our approach is solely based on edge orientation information. We have developed a method called Edge Orientation Matching (EOM) which we describe in this paper.



Fig. 1. Example of an edge orientation vector field computed using equ. (7).

2 Extraction of Edge Orientation Information

The extraction of edge information (strength and orientation) from a 2-D array of pixels $I(x, y)$ (a grey-scale image) is the basic feature calculation in our detection framework. In this work we use the Sobel method (see for example [3]) for edge processing. It is a gradient-based method which needs to convolve the image $I(x, y)$ with two 3×3 filter masks, K_x for horizontal filtering and K_y for vertical filtering. The convolution of the image with the two filter masks gives two edge strength images $G_x(x, y)$ and $G_y(x, y)$,

$$G_x(x, y) = K_x \star I(x, y), \quad (1)$$

$$G_y(x, y) = K_y \star I(x, y). \quad (2)$$

The absolute value $S(x, y)$, referred to as edge strength and the edge direction information $\Phi(x, y)$ are obtained using:

$$S(x, y) = \sqrt{G_x^2(x, y) + G_y^2(x, y)}, \quad (3)$$

$$\Phi(x, y) = \arctan\left(\frac{G_y(x, y)}{G_x(x, y)}\right) + \frac{\pi}{2}. \quad (4)$$

The edge information on homogenous parts of the image where no grey value changes occur is often noisy and bears no useful information for the detection. To exclude this information we apply a threshold T_s to the edge strength $S(x, y)$ generating an edge strength field $S_T(x, y)$,

$$S_T(x, y) = \begin{cases} S(x, y) & \text{if } S(x, y) > T_s \\ 0 & \text{else} \end{cases}. \quad (5)$$

The edge direction as stated in equation (4) takes on values from 0 to 2π . The direction of an edge depends on whether the grey value changes from dark to bright or vice versa. This information is irrelevant for our purposes. Therefore,

we map the direction information to a range of value $[0 \dots \pi]$ obtaining a new field

$$\hat{\Phi}(x, y) = \begin{cases} \Phi(x, y) & \text{if } 0 \leq \Phi(x, y) < \pi \\ \Phi(x, y) - \pi & \text{if } \pi \leq \Phi(x, y) < 2\pi \end{cases}, \quad (6)$$

which we call the edge orientation field. The edge orientation information can be rewritten using a complex formula

$$\mathbf{V}(x, y) = S_T(x, y)e^{j\hat{\Phi}(x, y)}, \quad (7)$$

where $\mathbf{V}(x, y)$ is the complex edge orientation vector field and $j^2 = -1$. $S_T(x, y)$ and $\hat{\Phi}(x, y)$ are obtained using equation (5) and (6). The edge orientation vector field can be displayed like shown in figure 11. The elements of \mathbf{V} are referred to as vectors \mathbf{v} .

3 Edge Orientation Matching

To build a face model, we use a sample of hand-labeled face images. The faces are cropped, aligned and scaled to the size 32×40 in the grey level domain. From this set of normalized face images an average face is computed. We also add vertically mirrored versions of each face in the set to the average face. Finally the edge orientation vector field $\mathbf{V}_M(x, y)$ is calculated from the average face. This is used as a model for the detection process. For face detection the model $\mathbf{V}_M(x, y)$ is shifted over the image, and at each image position (x, y) the similarity between the model and the underlying image patch is calculated. The image is represented by its orientation field $\mathbf{V}_I(x, y)$. In general the orientation matching process can be described as a convolution-like operation as

$$\mathbf{C}(x, y) = \sum_n \sum_m \text{dist}(\mathbf{V}_M(m, n), \mathbf{V}_I(x + m, y + n)), \quad (8)$$

where $\mathbf{C}(x, y)$ is an image like structure containing the similarity score between a sub-image of size $m \times n$ and the model which is of the same size for each possible model position within the image. The function $\text{dist}()$ calculates the local distance between two single orientation vectors. In order to find a face position $\mathbf{C}(x, y)$ is searched for all values that fall below a predefined matching threshold T_f . The local distance function $\text{dist}()$ is defined as a mapping of two 2-dimensional vectors \mathbf{v}_m and \mathbf{v}_i to $[0 \dots s_{max}]$. In our case they stem from an edge orientation field of the image \mathbf{V}_I and that of the model \mathbf{V}_M . They have the property $\arg\{\mathbf{v}\} = [0 \dots \pi]$. The upper bound for the distance s_{max} occurs when the vectors are perpendicular and both of maximal length. The value of s_{max} depends on the normalization of the vectors $\mathbf{v}_i, \mathbf{v}_m$. As we use 8-bit grey level-coded images we normalize the vectors so that we get $s_{max} = 255$. We use the following distance metric in the edge orientation domain:

$$\text{dist} = \begin{cases} \sin(|\arg\{\mathbf{v}_i\} - \arg\{\mathbf{v}_m\}|) \cdot s_{max} & \text{if } |\mathbf{v}_i|, |\mathbf{v}_m| > 0 \\ s_{max} & \text{else} \end{cases} \quad (9)$$

In [1] we introduced more ways to compute the function $\text{dist}()$. There we also propose an elastic matching method.

4 Fast Image Search

In order to find faces of different sizes use a resolution pyramid of edge orientation fields. The size ratio R_p between two resolution levels is set to $R_p = 1/25$. Each level of the pyramid is searched in a coarse to fine manner. First the similarity map \mathbf{C} is computed on a sparse grid with a grid step $D_{g^0} = 6$. So we approximately evaluate only 2.8% of all possible locations at this coarse level. Each grid point with a value below a threshold T_{gs} is a starting point for a local refinement of the search. In the next step we search at the points of a finer structured local grid around the starting point of the coarse grid. The grid step $D_{g^1} = 3$, which means that every 9th point is evaluated. Finally each point of that second step below T_{gs} is evaluated in a 3×3 neighborhood. The grid step of this third level is set to $D_{g^2} = 1$. Figure 2 illustrates the whole procedure. Points with a light grey tone are meant to be below the threshold T_{gs} . The speedup is

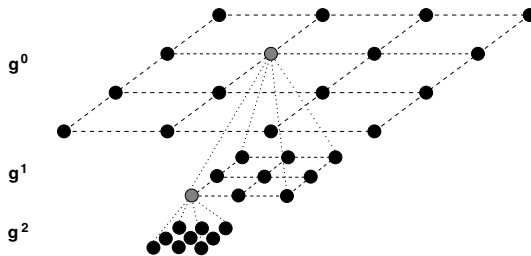


Fig. 2. Example of the hierarchical grid search.

controlled by the threshold T_{gs} which determines the number of grid points that are starting points for a finer analysis in their neighborhood. The higher T_{gs} is the less likely it will be that we have to perform fine grid searches. On the other hand it is clear that we are about to miss many true face positions then and the detection rate will decrease.

5 Experiments and Results

The image database used for testing the capabilities of the face detector consists of 17004 images, each showing at least one upright face. The whole data set consists of 4 different parts as listed in table 1. The images from the first three sets vary a lot with respect to image quality, lighting conditions, background and the size of the faces. The last set contains the images from the M2VTS [10] database which have an uniform background and the size of the displayed faces is alike. We regard a face as detected if the found face position and size do not deviate from the true values for more than a predefined small tolerance. The face position and size are defined by the position of the two eyes. In this work we count a face as detected if the eye position and eye distance do not deviate from

Table 1. Dataset used for the evaluation of our algorithm.

Name	#Images	#People	Comment
Test Set I	10775	210	complex background
Test Set II	1468	60	complex background, poor quality
Internet	2409	2409	complex background
m2vts	2352	295	simple background, high quality
Total	17004	2974	

the true values more than 20% in terms of the true eye distance. All the results reported in table 2 are obtained using a resolution pyramid with 6 resolution levels. The level size of the biggest level is 320×240 . The processing time less than 0.08 seconds on a Pentium II 500 MHz, using a Model which is of spatial size 32×40 and consists of 390 orientation vectors. So the proposed matching algorithm is suitable for real-time processing of video streams at about 12 frames per second.

Table 2. Detection results on our test dataset.

Dataset	Detection	#False Detects
Test Set I	95.5%	36954
Test Set II	75.5%	4573
Internet	93.3%	4530
m2vts	96.9%	71
Total	93.7%	41598

6 Related Work

There are several algorithms that use edge orientation information for face processing. Bichsel [1] uses dot products of orientation maps for purposes of face recognition. Burl et al. [2] utilize shape statistics of the facial feature (eyes, nose, mouth) configuration learned from a training sample. The feature candidates are detected by Orientation Template Correlation. Maio and Maltoni [4] utilize the hough transform for elliptical face outline detection with a subsequent matching of a manual generated orientation model for verification. The above mentioned methods are limited in the size of the detectable faces and the published results are obtained using only small databases.

7 Conclusions

In this work we show that edge orientation information is a powerful local feature for object detection. We achieve detection rates of more than 93% on a data base of 17000 test images. We also could show that the method is capable of real-time face detection on a standard PC. One main focus of our ongoing work now is to reduce the number of false detects.

Acknowledgments

The work described here was supported by the German Federal Ministry of Education and Research (BMBF) under the project EMBASSI.

References

1. Martin Bichsel. *Strategies of Robust Object Recognition for the Automatic Identification of Human Faces*. PhD thesis, Eidgenössische Technische Hochschule Zürich, Zürich, 1991.
2. M.C. Burl and P. Perona. Recognition of planar object classes. In *Proc. CVPR'96*, 1996.
3. Kenneth R. Castleman. *Digital Image Processing*. Prentice Hall, 1996.
4. Douglas Chai and King N. Ngan. Locating facial region of a head-and-shoulder color image. In *International Conference on Face and Gesture Recognition*, pages 124–129, 1998.
5. Beat Fasel. Fast multi-scale face detection. IDIAP-COM 4, IDIAP, 1998.
6. Bernhard Fröba and Christian Küblbeck. Face detection and tracking using edge orientation information. In *SPIE Visual Communications and Image Processing*, pages 583–594, January 2001.
7. Dario Maio and Davide Maltoni. Real-time face location on gray-scale static images. *Pattern Recognition*, 33:1525–1539, September 2000.
8. Stephen McKenna, Shaogang Gong, and Yogesh Raja. Face recognition in dynamic scenes. In *British Machine Vision Conference*, number 12, 1997.
9. K. Messer, J. Matas, J. Kittler, J. Luetttin, and Maitre G. Xm2vtsdb: The extended m2vts database. In *Second International Conference on Audio- and Video-based Biometric Person Authentication*, pages 71–77, 1999.
10. Henry A. Rowley. *Neural Network-Based Face Detection*. PhD thesis, Carnegie Mellon University, Pittsburgh, 1999.
11. Henry Schneiderman. *A Statistical Approach to 3D Object Detection Applied to Faces and Cars*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, May 2000.
12. Q.B. Sun, W.M. Huang, and J.K. Wu. Face detection based on color and local symmetry information. In *International Conference on Face and Gesture Recognition*, pages 130–135, 1998.
13. Jean-Christophe Terrillon, Martin David, and Shigeru Akamatsu. Automatic detection of human faces in natural scene images by use of a skin color model and of invariant moments. In *International Conference on Face and Gesture Recognition*, pages 112–117, 1998.
14. Jie Yang, Weier Lu, and Alex Waibel. Skin-color modelling and adaption. In *ACCV'98*, 1998.
15. Ming-Hsuan Yang, Dan Roth, and Narendra Ahuja. A snow-based face detector. In *Advances in Neural Information Processing Systems 12 (NIPS 12)*, pages 855–861. MIT Press, 2000.

Directional Properties of Colour Co-occurrence Features for Lip Location and Segmentation

Samuel Chindaro and Farzin Deravi

Electronics Department, University of Kent at Canterbury
Canterbury, Kent, CT2 7NT, United Kingdom
{S.Chindaro,F.Deravi}@ukc.ac.uk

Abstract. Automatic lip tracking is based on robust lip location and segmentation. Here an algorithm which can locate the position of the lips robustly without the constraints of lip highlighting or special lighting conditions is proposed. The proposed method is based on the directional properties of the features of co-occurrence matrices to distinguish between facial parts. The method essentially consists of three parts: a) a face location module b) a facial features location module c) a feature identification module which identifies the lips. The proposed algorithm uses the hue information only in the HSV colour space. The method has been tested on the XM2VTS database, with a high success rate. The use of hue and textural features to do the processing makes the algorithm robust under various lighting conditions.

1 Introduction

Tracking of lip movements has attracted the attention of the research community because of the additional information conveyed to compliment audio-based speech and speaker recognition techniques [1,2,3]. Lip localisation is important as the first step in lip tracking and reading. Not only are lip movements used to aid in person recognition, but also lip signatures have been used for automatic person recognition [1]. In all cases, before any of the verification or identification process can take place, an approximate position of the lips must be established. The principal difficulty is that lips are set against flesh tones with weak contrast. A number of approaches have been proposed in literature, some based on grey level analysis, and others on colour analysis [4]. A number of these methods impose strong constraints such as blue make up [1] or adapted illumination [5]. Some of these constraints are hardly achievable for practical applications. There have been a number of approaches proposed which use intensity information to locate the lips, particularly using the grey level profile [6]. The drawback with intensity based methods is that they tend to be very sensitive to lighting variations. The method proposed in this paper is based on colour textural features and chromaticity information. The advantage in using this approach is that the procedure is then relatively insensitive to changes in absolute brightness.

Grey scale co-occurrence probabilities are an example of a feature-based texture segmentation method. The probabilities are represented as entries in a matrix of relative frequencies. The frequencies are counts which describe how often two pixels located at a fixed geometric position relative to one another have a particular pair of

grey levels. Haralick *et al* [7] have proposed a variety of secondary features that can be employed to extract useful texture information from the co-occurrence matrix (COM). Most of the work has focused on grey-level representation while considerably less work has been presented on how to combine chromatic (colour) with structural (texture) information [8].

The use of directional properties of colour co-occurrence matrices (CCOMs) to locate the face in grey-scale and colour images was presented in [9]. The method was first applied to grey-scale images, and an improvement was made when colour information was used. Based on the feature parameters of COMs, a face-texture model composed of a set of inequalities (directional properties) was derived. A face area was defined as a region in which these parameters hold. In this work, we present the use of the directional properties of colour co-occurrence matrices to distinguish the lips from the rest of the facial parts. Our work was carried out entirely in the hue space of the HSV colour space.

We used colour (hue only) to locate the face based on a skin pixel model extracted from the XM2VTS [10] database and the hue gradient profile to locate face feature borders. The XM2VTS face database contains four recordings of 295 subjects taken over a period of four months. 50 images that represent a cross-section of faces were extracted from the XM2VTS database and used as a training set. From this database facial features were manually extracted each of an identical size of 115 x 64. The features extracted were the nose, eyes, mouth and the chin. We modelled the lips texture features based on the directional properties of colour co-occurrence matrices, from face features manually extracted from the same database.

2 Co-occurrence Matrices

Co-occurrence matrices (COMs) count how often pairs of grey levels of pixels, that are separated by a certain distance and lie along a certain direction, occur in a digital image. The COM is constructed by observing pairs of image cells distance d from each other and incrementing the matrix position corresponding to the grey level of both cells. This allows us to derive four matrices for each given distance: $P(0^\circ, d)$, $P(45^\circ, d)$, $P(90^\circ, d)$ and $P(135^\circ, d)$. The matrices are normalised and derived features which capture some characteristics of textures such as, homogeneity, coarseness, periodicity and others are computed.

3 Colour Space

Colour information is commonly represented in the widely used RGB co-ordinate system. The difficulty of using the RGB space is that it is hardware oriented and is suitable for acquisition or display devices, but not particularly applicable in describing the perception of colour [11] and does not closely model the psychological perception of colour. Intensity is distributed throughout all three parameters, rendering colour values highly sensitive to lighting changes. The HSV (hue, saturation and value) model separates hue, saturation and intensity. In many cases, hue is invariant to shadows, shading and highlights. Important advantages of HSV and related colour spaces over other colour spaces are separability of chromatic values from achromatic

values and the possibility of using one feature (H) only for segmentation as we demonstrated in [4] and again in this paper. The HSV model is used in the method described in this paper.

4 Feature Modelling Based on CCOM

Experiments were carried out on the extracted facial parts, manually extracted from the database. The investigation involved extracting the following features from the co-occurrence matrices of these facial parts: energy, entropy, inertia, local homogeneity and correlation. Investigations showed that the most distinctive feature was the correlation feature, which is defined in (1). The shape of the mouth suggested the value of the correlation feature along the horizontal axis of the mouth (0°), is related to the same feature in the vertical direction (90°) by expression (2) where α is a constant that is determined experimentally. The mouth shape also suggests that the correlation feature at an angle of 45° is greater than that in the vertical direction, expressed as in (3).

$$C_r = \sum_i \sum_j (ij) p(i,j) - u_x u_y / \delta_x \delta_y \quad (1)$$

Where $p(i,j)$ is the (i,j) th entry in a normalised colour COM. u_x, u_y and δ_x, δ_y are the means and standard deviations of the row and column sums of the matrix[7].

$$Cr90 + \alpha > Cr0 \quad (2)$$

$$Cr45 > Cr90 \quad (3)$$

This hypothesis was tested on the 200 samples (50 of each part) which we had extracted. The inter-pixel distance was also varied. Table 1 summarises the results obtained.

Table 1. Results of tests carried out on various facial parts.

No of Images	Facial Part	No of facial parts identified as lips							
		Inter-pixel Distance							
		<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>10</i>	<i>12</i>	<i>14</i>
50	Lips	50	50	50	48	47	49	48	46
50	Noses	3	3	4	4	2	3	3	3
50	Eyes	24	16	13	13	10	10	10	10
50	Chins	3	3	3	3	3	2	2	2

The results showed that the above hypothesis held for 100% of the lip images up to a distance of 3. As the inter-pixel distance was increased, some mis-classified parts were eliminated. At larger inter-pixel distances of 10 and above, we began to get some false positives. The results for the eyes showed that due to the similar oval shape, a number of eyes were identified as lips (26%). But this was not a very significant factor as other factors like position and colour could easily be used to eliminate the eyes. In fact our algorithm eliminates eyes in most cases by assuming that the mouth lies in the lower three-quarters of the face, and that it is almost central. An inter-pixel distance of 3 was subsequently used since this was the distance at which we got 100% lips identification.

5 Face Localisation and Possible Candidates Segmentation

The original full image is first reduced into a region of interest (ROI) image that is comprised of mostly the face region. The colour of skin in images depends mainly on the concentration of haemoglobin and melanin and on the conditions of illumination. It has widely been reported that the hue of the skin is roughly invariant across different ethnic groups after the illumination has been discounted [12]. This is because differences in the concentration of pigments primarily affect the saturation of skin colour, not the hue. We therefore use hue only to localise the face. First likely skin pixels are labelled using the mean hue (0.062) obtained from the skin database¹. The image is then scanned horizontally and vertically to locate connected labelled pixels. A grouping of between 30-50 connected components was found to be adequate to give a good approximation of the face borders. The first approximation of the mouth area is the lower three-quarters of the localised face.

The location of the mouth is first approximated to be roughly at the centre, horizontally. From either side of the centre 15 columns² are extracted (a total of 31 columns). The mean hue values of these pixels are found. The gradient profile is then constructed.

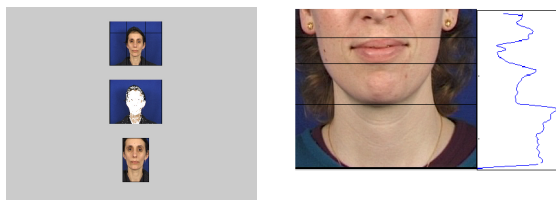


Fig. 1. (left) Face Localisation and first estimate of mouth region (right) Candidates border location using hue gradient profile.

¹ Hue is mapped from 0-360 degrees to between 0 and 1; using Matlab 5.2.

² These values were empirically found to give the best results.

This is done by taking the modulus of the difference between adjacent pixels. This profile is then filtered using an averaging filter of size 5. The first 10 maxima on the gradient profile are identified. These are then clustered into four clusters using k-means clustering. The cluster centre positions are marked as cues to possible lip boundary locations. The positions of maximum hue gradient are marked as cues to lip/skin boundary location. From the centre along the x-axis on each mouth candidate, an area of $30 \times 15 \text{ pixels}^2$ is selected on either side of the marked positions for further processing by the COM directional classifier module.

6 Lip Localisation Using CCOM Model

The identification of the lips is done using the CCOM directional classifier module. The extracted candidates (see fig.2) were tested for directional conditions depicted by inequality (3).



Fig. 2. Extracted Candidates.

7 Results

The correct classification of the lips depended on the successful isolation of the lips using the candidates isolation module. Out of the 50 facial images tested, the mouth was successfully isolated in 40 of the images, a success rate of 80%. Out of these the mouth was correctly identified in 38 of the images, a rate of 95%. Images which failed in the candidates isolation module included face with black or white moustaches or beards and faces with very thin lips. Once the candidates were successfully segmented the classification rate for the mouth using the colour COM directional classifier was very high.

8 Conclusion

We have presented an innovative approach of combining colour information and the directional properties of colour COM with application to lip segmentation. Only one colour component, hue, was used to carry out the procedures which utilised the chromatic and the textural information in this plane. Given a module which can successfully locate facial features without identification, we have proven that CCOMs can be successfully used to identify the lips, which are an important feature for complimenting audio-based speech and speaker recognition techniques. This work can further be developed by carrying out investigations on the textural properties of the facial features such as the eyes, for other biometrics application

References

1. Auckenthaler, R, Brand, J, Mason J.S., Deravi, F., and Chibelushi, C.C. Lip Signatures for Automatic Person Recognition 2nd International Conference on Audio and Video-based Biometric Person Authentication, March 22-23, 1999, Washington, DC, USA, pp. 142-147.
2. Matthews, I., Bangham, J.A., and Cox, S. Scale Based Features For Audio-Visual Speech Recognition, IEE Colloquium on Integrated Audio-Visual Processing for Recognition, Synthesis and Communication, London, Nov. 1996.
3. Sanchez, M., Matas, J, and Kittler J. Statistically chromaticity model for lip-tracking with B-splines. AVBPA 97.
4. Chindaro S. and Deravi F. Lip Localisation Using Chromaticity Information Proc. 1st Conf. In Colour in Graphics and Image Processing , Saint-Etienne, France 00. pp. 343-347.
5. Vogt M. Interpreted multi-state lip models for audio-video speech recognition In Proc. Of the Audio-Visual Speech Processing, Coignitive and Computational Approaches Workshop, ISSN 1018-4554, Rhodes,Greece 1997.
6. Delmas P., Coulon P.Y.,and Fristot V., Automatic Snakes for Robust Lip Location IEEE Int. Conf. On Acoustics Speech, and Signal Processing, Proceedings, 15-19 March 1999, pp. 3069-3072.
7. Haralick R.M, Shanmugam K., and Dinstein I. Textural Features For Image Classification IEEE Trans. On Syst., Man and Cybernetics, vol. Smc-3,, no. 6, pp. 610-621, 1973.
8. Paschos G. Chromatic correlation Features for Texture Recognition Pattern Recognition Letters Vol. 19 ,1998 pp. 643-650.
9. Dai Y. and Nakano Y. Face-Texture Model Based on SGLD an its Application In Face Detection in a Colour Scene Pattern Recognition, Vol. 29, no, pp. 1007-1017, 1996.
10. The XM2VTS Face Database web site,
<http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb/>.
11. Foely, J., VAN Dam, A., Feiner, S., and Hughes, J. Computer Graphics: Principals and Practice Addison Wesley, Reading, MA, 1990.
12. M.J. Jones and J.M. Rehg Statistical Colour Models with Application To Skin Detection Cambridge Research Laboratory, Technical Report Series, Compaq, (Dec. 1998).

Robust Face Detection Using the Hausdorff Distance

Oliver Jesorsky, Klaus J. Kirchberg, and Robert W. Frischholz

BioID AG, Berlin, Germany

{o.jesorsky,k.kirchberg,r.frischholz}@bioid.com

<http://www.bioid.com>

Abstract. The localization of human faces in digital images is a fundamental step in the process of face recognition. This paper presents a shape comparison approach to achieve fast, accurate face detection that is robust to changes in illumination and background. The proposed method is edge-based and works on grayscale still images. The *Hausdorff distance* is used as a similarity measure between a general face model and possible instances of the object within the image. The paper describes an efficient implementation, making this approach suitable for real-time applications. A two-step process that allows both coarse detection and exact localization of faces is presented. Experiments were performed on a large test set base and rated with a new validation measurement.

1 Introduction

Face recognition is a major area of research within biometric signal processing. Since most techniques (e.g. Eigenfaces) assume the face images normalized in terms of scale and rotation, their performance depends heavily upon the accuracy of the detected face position within the image. This makes face detection a crucial step in the process of face recognition.

Several face detection techniques have been proposed so far, including motion detection (e.g. eye blinks), skin color segmentation [5] and neural network based methods [3]. Motion based approaches are not applicable in systems that provide still images only. Skin tone detection does not perform equally well on different skin colors and is sensitive to changes in illumination.

In this paper we present a model-based approach that works on grayscale still images. It is based on the Hausdorff distance, which has been used for other visual recognition tasks [4]. Our method performs robust and accurate face detection and its efficiency makes it suitable for real-time applications.

2 Hausdorff Object Detection

The Hausdorff distance (HD) is a metric between two point sets. Since we want to use it for object detection in digital images, we restrict it to two dimensions.

2.1 Definition

Let $\mathcal{A} = \{a_1, \dots, a_m\}$ and $\mathcal{B} = \{b_1, \dots, b_n\}$ denote two finite point sets. Then the Hausdorff distance is defined as

$$H(\mathcal{A}, \mathcal{B}) = \max(h(\mathcal{A}, \mathcal{B}), h(\mathcal{B}, \mathcal{A})) , \quad \text{where} \quad (1)$$

$$h(\mathcal{A}, \mathcal{B}) = \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} \|a - b\| . \quad (2)$$

Hereby $h(\mathcal{A}, \mathcal{B})$ is called the *directed Hausdorff distance* from set \mathcal{A} to \mathcal{B} with some underlying norm $\|\cdot\|$ on the points of \mathcal{A} and \mathcal{B} .

For image processing applications it has proven useful to apply a slightly different measure, the (directed) *modified Hausdorff distance* (MHD), which was introduced by Dubuisson et al. [11]. It is defined as

$$h_{\text{mod}}(\mathcal{A}, \mathcal{B}) = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} \|a - b\| . \quad (3)$$

By taking the average of the single point distances, this version decreases the impact of outliers making it more suitable for pattern recognition purposes.

2.2 Model-Based Detection

Rucklidge [4] describes a method that uses the HD for detecting an object in a digital image. Let the two-dimensional point sets \mathcal{A} and \mathcal{B} denote representations of the image and the object. Hereby, each point of the set stands for a certain feature in the image, e.g. an edge point. The goal is to find the transformation parameters $p \in \mathcal{P}$ such that the HD between the transformed model $T_p(\mathcal{B})$ and \mathcal{A} is minimized (see fig. 1). The choice of allowed transformations (e.g. scale and translation) and their parameter space \mathcal{P} depends on the application. Efficient HD calculation allows an exhaustive search in a discretized transformation space.

The detection problem can be formulated as

$$d_{\hat{p}} = \min_{p \in \mathcal{P}} H(\mathcal{A}, T_p(\mathcal{B})) . \quad (4)$$

Then we call $h(T_p(\mathcal{B}), \mathcal{A})$ the *forward distance* and $h(\mathcal{A}, T_p(\mathcal{B}))$ the *reverse distance*, respectively. To consider only that part of the image which is covered by the model, we replace the reverse distance by the *box-reverse distance* h_{box} . The definition can be found in [4].

3 System Description

The implemented face detection system basically consists of a coarse detection and a refinement phase, each containing a segmentation and a localization step. The following sections discuss these two phases in detail. Figure 2 gives a general overview of the described system.

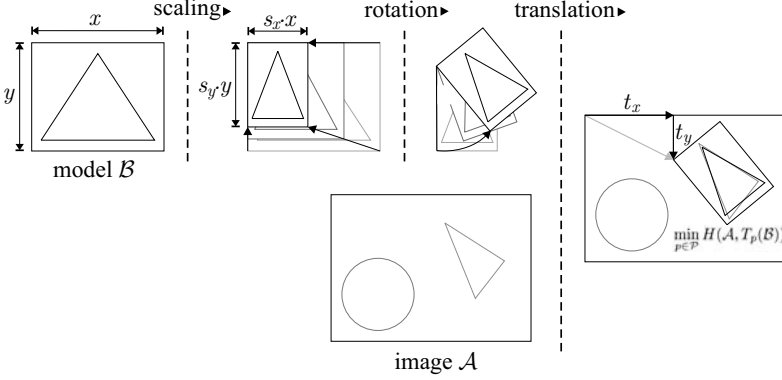


Fig. 1. Model fitting by scaling, translation and rotation.

Coarse Detection: Before applying any segmentation step, an area of interest (AOI) with preset width/height ratio is defined for each incoming image f . This AOI is resampled to a fixed size to be independent of the dimensions of f .

- *Segmentation:* An edge intensity image is calculated from the resized AOI with the Sobel operator. Afterwards, local thresholding guarantees that the resulting binary edge points are equally distributed over the whole image area.
- *Localization:* With a face model \mathcal{B} and the binary representation \mathcal{A} obtained by the segmentation step, a localization of the face in the image can now be performed according to equation (4). Experiments have proven that the modified forward distance is sufficient to give an initial guess for the best position. The parameter set \hat{p} that minimizes $h(T_p(\mathcal{B}), \mathcal{A})$ is used as input for the refinement phase.

Refinement: Based on the parameter set \hat{p} , a second AOI is defined covering the expected area of the face. This area is resampled from the original image f resulting in a grayscale image h of the face area. Segmentation and localization are equivalent to the coarse detection step, except that a more detailed model \mathcal{B}' of the eye region is used.

The values of the modified box reverse distance $h_{\text{box}}(\mathcal{A}', T_{\hat{p}'}(\mathcal{B}'))$ at the best position, especially when multiplied with the modified forward distance $h(T_{\hat{p}'}(\mathcal{B}'), \mathcal{A}')$, can be used to rate the quality of the estimation. This is helpful if a face/non-face decision is desired. The eye positions are calculated from the parameter sets \hat{p} and \hat{p}' . Compared with manually set eye positions they are used to rate the quality of the system.

If the localization performance is not sufficient, an exact determination of the pupils can be achieved by additionally applying a multi-layer perceptron (MLP) trained with pupil centered images. The MLP localizer is not discussed in detail, but the results gained in combination with the rest of the system are included in the result section.

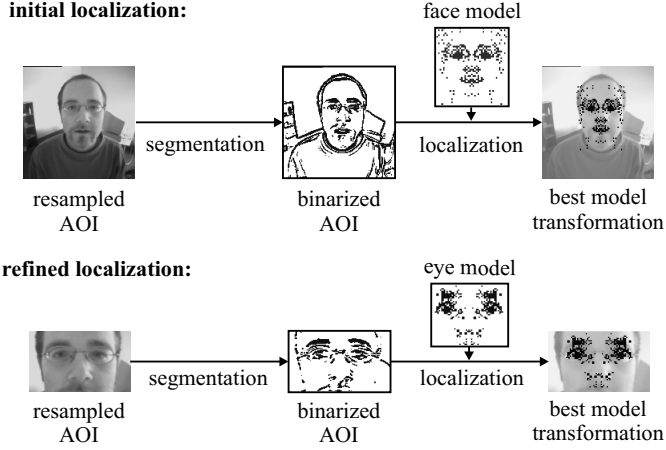


Fig. 2. The two phases of the face detection system containing the segmentation and the localization steps (AOI = area of interest). Top: coarse detection with a face model; bottom: refinement of the initially estimated position with an eye model.

Model Choice: The face and eye models, which are shown in figure 2, were initialized with average face data and optimized by genetic algorithms on a test set containing more than 10000 face images.

4 Validation

To validate the performance of our face detection system we introduce a relative error measure based on the distances between the expected and the estimated eye positions.

We use the maximum of the distances d_l and d_r between the true eye centers $C_l, C_r \in \mathbb{R}^2$ and the estimated positions $\tilde{C}_l, \tilde{C}_r \in \mathbb{R}^2$ as depicted in figure 3a. This distance is normalized by dividing it by the distance between the expected eye centers, making it independent of scale of the face in the image and image size:

$$d_{eye} = \frac{\max(d_l, d_r)}{\|C_l - C_r\|}. \quad (5)$$

In the following we will refer to this distance measure as *relative error*.

Considering the fact that in an average face the distance between the inner eye corners equals the width of a single eye, a relative error of $d_{eye} = 0.25$ equals a distance of half an eye width, as shown in figure 3b.

5 Experimental Results

Experiments on different test sets with different resolutions and different lighting and background conditions have been performed. The distribution function of the relative error between the expected eye positions (manually set) and the

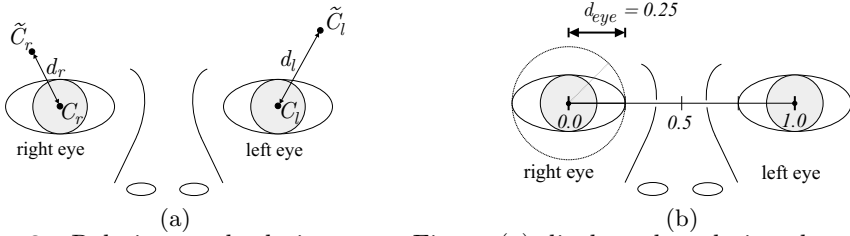


Fig. 3. Relations and relative error. Figure (a) displays the relations between expected (C_l and C_r) and estimated eye positions (\tilde{C}_l and \tilde{C}_r); (b) shows the relative error with respect to the right eye (left in image). A circle with a radius of 0.25 relative error is drawn around the eye center.

positions after each processing step has been calculated for each set. In this paper we present the results calculated on two test sets.

First one is the commonly used **extended M2VTS database** (XM2VTS) [2]. This database contains 1180 color images, each one showing the face of one out of 295 different test persons. Before applying the HD face finder we converted the images to grayscale and reduced their dimension to 360×288 pixel.

The second test set, which we will refer to as the **BIOID database**, consists of 1521 images (384×288 pixel, grayscale) of 23 different persons and has been recorded during several sessions at different places of our company headquarters. Compared to the XM2VTS this set features a larger variety of illumination, background and face size.

To give all researchers the opportunity to compare their results with ours, this test set is available for public at www.bioid.com/research/index.html (including manually set eye positions).

A comparison of some images of the two test sets can be seen in figure 4.

For our experiments we allowed translation and scaling. The search area was restricted to a square, horizontally centered region covering the whole image height. The scaling parameter range was chosen such that faces taking up between 25% and 55% of the image width could be detected. Figure 4 shows the

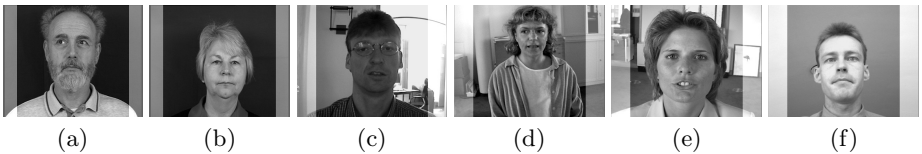


Fig. 4. Image samples from XM2VTS (a, b) and the BIOID test set (c, d, e, f). Areas not considered during search have been brightened.

resulting distribution functions for both test sets. The results for HD search with additional MLP refinement have been included in the graphs.

If we consider a face found if $d_{eye} < 0.25$, the XM2VTS set yields 98.4% and the BIOID set 91.8% of the faces localized after refinement. This bound allows a maximum deviation of half an eye width between expected and estimated eye position (fig. 3b) as described in section 4.

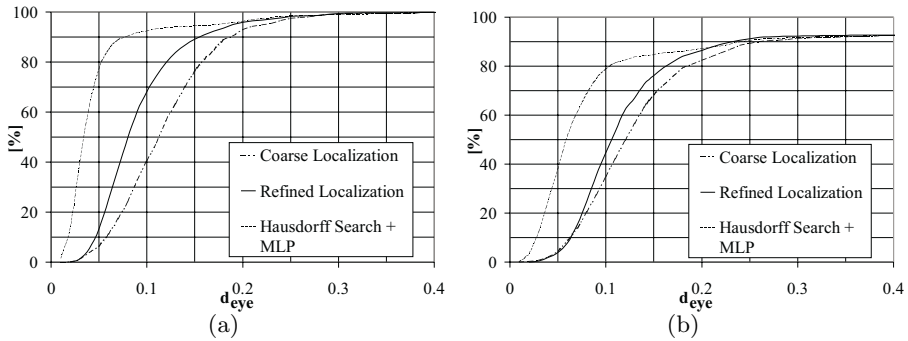


Fig. 5. Distribution function of relative eye distances for the XM2VTS (a) and the BIOD test set (b).

The average processing time per frame on a PIII 850 MHz PC system is 23.5 ms for the coarse detection step and an additional 7.0 ms for the refinement step, which allows the use in real time video applications (> 30 fps).

6 Conclusions and Future Research

In this paper we presented a face detection system that works with edge features of grayscale images and the modified Hausdorff distance. After a coarse detection of the facial region, face position parameters are refined in a second phase.

System performance has been examined on two large test sets by comparing eye positions estimated by the system against manually set ones with a relative error measure that is independent of both the dimension of the input images and the scale of the faces. The good localization results show that the system is robust against different background conditions and changing illumination. The runtime behavior allows the use in realtime video applications.

Future research will concentrate on abolishing the restrictions of the detection of only frontal views and single faces, on automatic model creation and on transformation parameter optimization.

References

- [1] M.P. Dubuisson and A.K. Jain. A modified Hausdorff distance for object matching. In *ICPR94*, pages A:566–568, Jerusalem, Israel, 1994.
- [2] K. Messer, J. Matas, J. Kittler, J. Luetten, and G. Maitre. XM2VTSDB: The extended M2VTS database. In *Second International Conference on Audio and Video-based Biometric Person Authentication*, pages 72–77, March 1999.
- [3] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 203–207, San Francisco, CA, 1996.
- [4] W. Rucklidge. *Efficient Visual Recognition Using the Hausdorff Distance*, volume 1173 of *Lecture notes in computer science*. Springer, 1996.
- [5] J. Terrillon, M. David, and S. Akamatsu. Automatic detection of human faces in natural scene images by use of a skin color model and of invariant moments. In *Proc. of the Third International Conference on Automatic Face and Gesture Recognition*, pages 112–117, Nara, Japan, 1998.

Multiple Landmark Feature Point Mapping for Robust Face Recognition

Menaka Rajapakse and Yan Guo

RWCP*, Multi-Modal Functions KRDL**
21 Heng Mui Keng Terrace, Singapore 119613
menaka@krdl.org.sg

Abstract. This paper presents a technique to identify faces using correspondence of feature points between faces. One to many correspondence mapping among feature points is performed to achieve the best fit among feature points of two given face images. A set of nine feature points is selected during image registration, which represents the approximated landmark points of the set of face images of the registered person. During recognition, a 5x5 neighborhood of the matching image anchored at each corresponding feature point location in the registered image is searched for the best matching point. As a result, a set of feature vectors for the matching image is secured. Feature vectors are calculated using Gabor responses at these points as they are known to be effective in providing local feature descriptors. The best estimation for matching points and the final face similarity is calculated using the nearest-neighbor algorithm using the Euclidean distance. We evaluate the effectiveness of the feature selection method described in this paper on frontal and near-frontal face images in a large database.

1 Introduction

The techniques employing 2D filters to extract image features have been effective compared to template based explicit feature mappings for face recognition. Gabor-based wavelet approaches have recently been advocated and achieved successful results [1] [2]. Factors such as change of pose, facial expressions add ambiguities making automatic face recognition a difficult task. In this paper we investigate how a wavelet based representation responds to recognizing frontal face images when the registered images are all frontal or frontal and near-frontal images. We will test our method on two sets of images categorized based on their pose; frontal, near-frontal. The image database consists of four images per person and are normalized for scale, rotation, translation and illumination invariance. The landmark feature points are manually selected during face registration and these point locations act as anchor points for feature point mapping on the images to be recognized. For our experiment we select a total of nine feature points

* Real World Computing Partnership.

** Kent Ridge Digital Labs.

located on facial landmarks such as eyes, nose and mouth. Nearest neighbor Euclidean matching is used to select the best mapping of the feature points and for face matching. The method described in this paper is currently tested only on static images but has the potential of extending to dynamic sequences.

2 Image Normalization

Robustness of a face recognition algorithm heavily relies on the accuracy of the preprocessing step. Normalization of the face image for scale, rotation, translation and illumination invariance is crucial for accurate feature extraction. The normalization method used in our approach is based on the located eye positions [3]. In order to achieve faces invariant to rotation, translation and scale, a transformation matrix is computed by joining the located eye positions, that yields a horizontal segment having a length of 52 pixels separating the two eyes. The normalized output image is of size 150x200 with the positions of the eyes fixed at preset coordinates and has similar grey levels.

3 Face Registration

The face registration is carried out manually by mouse clicking at the feature points. The feature points of interest are two eye centers, three points on the nose base and finally four points on the mouth encapsulating the profile of the mouth(left and right corners and top and bottom points) of face images. Three images per person are used during the image registration, and the selection of these images is carried out as follows: all frontal faces; combination of frontal and near-frontal faces.

4 Gabor Wavelet Representation

The 2-D complex valued Gabor kernel used for retrieving multi-scale and multi-orientation features [4] [5] is given by:

$$\psi_k(z) = \frac{k^2}{\sigma^2} \exp\left(-\frac{k^2 z^2}{2\sigma^2}\right) [\exp(ikz) - \exp\left(-\frac{\sigma^2}{2}\right)] \quad (1)$$

The Gabor wavelets are generated for three frequencies and six orientations. These generated filter kernels are then used to extract Gabor features. Gabor features at each spatial location are extracted by taking the convolution of the grey level values in the neighborhood window of the feature point with the family of different Gabor wavelets generated at different scales and orientations.

5 Feature Point Selection and Mapping

The classification decision is influenced by the components of feature vectors evaluated at feature points, and the saliency of a feature point can be evaluated

by measuring the magnitude of each component of the feature vector. Saliency maps have been developed for facial expression analysis by taking the discriminant vector component magnitude averaged over all orientations, scales, and facial expressions at various feature points [6]. And the results show that the eyes and mouth are the most crucial regions for determining facial expressions. Although our main focus in this paper is to recognize faces and not facial expressions, we opt for feature points which conform with above findings as our database contains faces with varying facial expressions.

The feature points used in our experiment are illustrated in Figure 1. A face $F = \{f_i : i = 1 \cdots 9\}$ is represented by a set of features and f_i is a vector representing magnitudes of multiscale Gabor wavelet filters extracted at the i^{th} feature. The total number of feature points selected is nine.

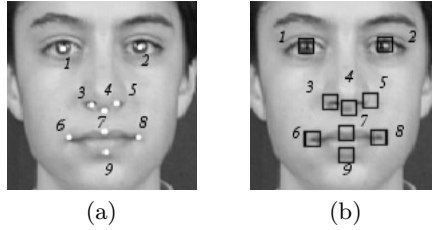


Fig. 1. (a)Registration of feature points (b)Regions for feature point mapping for recognition.

Given a large database of images, to establish the identity of an image it is required to determine any relationship or similarity between the two images in comparison. In our approach, the most similar point on the matching image corresponding to a predetermined point on the registered image is sought. In order to accomplish this, each feature point on the registered image is mapped to a data field of a 5×5 window in the neighborhood of an anchored point on the matching image. These anchor points are the feature point locations of the registered image. The most similar feature point based on the nearest neighbor classification is considered as the best match for the point under consideration. Similarly, correspondence for all points on the registered image is established. Once all correspondence are established, these chosen locations on the matching image are presumed to be the locations of its respective landmark feature points. Let the registered face image be $F_R = \{f_{R_1}, \dots, f_{R_9}\}$, test image on which feature points are to be mapped $F_T = \{f_{T_1}, \dots, f_{T_9}\}$, and the resulting mapped image be $F_M = \{f_{M_1}, \dots, f_{M_9}\}$ where $f_{R_i} = (x_{f_{R_i}}, y_{f_{R_i}})$, $f_{T_i} = \{(x_{f_{T_j}}, y_{f_{T_j}}) : j \in N_i\}$, N_i represents the 5×5 neighborhood window and $f_{M_i} = (x_{f_{M_i}}, y_{f_{M_i}})$. A minimum distance classifier with Euclidean distance is used to calculate the best match for each feature point.

$$(x_{f_{M_i}}, y_{f_{M_i}}) = \arg \min_{j \in N_i} d\{(x_{f_{R_i}}, y_{f_{R_i}}), (x_{f_{T_j}}, y_{f_{T_j}})\}$$

6 Face Classification

A given face is said to be recognized when it is successfully matched with a face in the database. Explained below is the face classification procedure utilized for choosing the best match for a given face image. The magnitude of the complex Gabor response at each feature point f_i is evaluated by convolving the grey value of the feature point with 18 Gabor kernels. Hence, we can represent a feature vector corresponding to a particular feature point as a collection of Gabor coefficients: $f_i = (f_{i1}, f_{i2}, \dots, f_{i18})^T$ where f_{ij} corresponds to the Gabor feature j evaluated for the facial feature vector i and eighteen elements correspond to three spatial frequencies and six orientations.

Feature vectors each consisting of 18 Gabor response amplitudes are generated for each predefined feature point of the registered image and the matching images in the database. The minimum distance between feature vectors of two mapped feature points on two images is determined by a nearest neighbor classifier.

The final confidence measure between two images is calculated by taking the average of all the distances between the corresponding feature points:

$$d_{f_i f_i'} = \min\{||f_i - f_i'||\}_{i=1 \dots N}$$

$$d_{FF'} = \left\{ \frac{\sum_{i=1 \dots N} d_{f_i f_i'}}{N} \right\}_{\min}$$

where $d_{f_i f_i'}$ is the distance between f_i and f_i' and $d_{FF'}$ is the distance between two faces F and F' and $N = 9$. A small distance between two images indicates that the two face images lie closer in the face space and therefore said to be similar in nature.

7 Face Database

The present system was tested on XM2VTSDB face database from University of Surrey which consists of 1180 images, with 4 images for person taken at four different time intervals (one month apart). Though similar lighting conditions and backgrounds have been used during image acquisition significant changes in hair styles, facial hair, glasses are present in images. These images consists of frontal and near frontal images with somewhat dissimilar facial expressions. The original image size is 726 x 576 pixels and contains images of Caucasian and Asian males and females.

8 Experiments and Results

In order to evaluate the effectiveness of the proposed feature selection technique across view pose variations, experiments were performed on two data sets. The

first data set contains 471 registered frontal images (3 per person), and 157 frontal images for matching. The second set consists of 414 registered frontal and near-frontal images, and 138 frontal images for matching. Resolutions of images used in the experiment are 128x128 pixels. Examples of the categorized face image samples are shown in Figure 2.



Fig. 2. (a)Left column: Matching Images, Right columns: Frontal Registered Images (b)Left column: Matching Images, Right columns: Frontal & near-frontal Registered Images.

Table 1. Performance measures.

View Pose	Matched Images	Registered Images	EER (%)	Recall rate at 100% precision
Frontal Only	157	471	4.20	96
Frontal + Near Frontal	138	414	8.42	85

The standard measures, namely *false acceptance rate* (FAR) and *false rejection rate* (FRR) were used to evaluate the effectiveness of the proposed system across slight pose variations. We have also quoted the *equal error rate* (EER) for the comparison of performance with existing algorithms. Firstly we tested out the frontal face image of the same face ensemble against their respective registered frontal images using “leave-one-out” scheme. This process was repeated four times, each time leaving a different sample out. Results were averaged to get the final measure. The second experiment was performed by comparing frontal face images against their pre-registered frontal and near-frontal counterparts with dissimilar facial expressions. Experiment results are illustrated in Table 1. As seen in Table 1, for frontal face ensemble, the true acceptances and rejections are 96% and the false rejections and acceptances are kept at a maximum of 4% which is quite reasonable. Moreover, the recall rate at 100% precision for face verification is 96%, which is the rate at which a true match is found in the first

position for the face being tested. The recognition rate of a frontal image from a registered set of frontal and near frontal image combination has dropped to an overall recognition rate of 92%, and the achieved recall rate is 85%. Furthermore, the discrimination capability of the proposed system deteriorates with the introduction of images with remarkable pose and expression variations.

9 Conclusion

We have explored an approach to recognize frontal faces by comparing them with corresponding pre-registered frontal and near frontal face images. One to many correspondence mapping among feature points is performed to achieve the best fit among feature points of two given face images. A minimal number of feature points was selected and the Gabor responses at those respective feature locations were collectively used as a suitable representation for a face. Similarity of the feature vectors in the data space was determined by the Euclidean distance. A useful user-cooperative face recognition system (with minimum pose and expression variations) is realized, and may be used effectively for non real-time applications. We have also demonstrated that nine feature points are sufficient to achieve an accuracy rate of (96%) for frontal face recognition when the registered faces are frontal. The accuracy of the system for near-frontal images can be improved by selecting more feature points as in [7] and by increasing the number of kernels used in the experiment. By doing so however, the overhead and the computational complexity will increase considerably making the proposed system impractical for real-time applications.

References

1. Manjunath, B.S., et al. "A Feature Based Approach to Face Recognition " *Proc. IEEE Computer Vision and Pattern Recognition* 1992, pp. 373-378.
2. Wiskott, L. et al., " Face Recognition by Elastic Bunch Graph matching", *Proc. IEEE International Conference on Image Processing*, Vol. 1 1997 pp. 129-132.
3. Huang, W. et al., "A robust approach to face and eye detection from images with cluttered background", *Proc. IEEE 14th International Conference on Pattern Recognition*, vol. 1, 1998, pp. 110-113.
4. Daugman, J.G., "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters", *Journal of Optical Society of America*, vol. 2, No. 7 July 1985.
5. Lee T.S., "Image Representation Using 2D Gabor wavelets", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, 1996.
6. Lyons M.J. et al., "Automatic Classification of Single Facial Images", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, 1999, pp. 1357-1362.
7. Rajapakse, M., Yan, G., " Performance Analysis of Gabor Responses in Face Recognition", *Proc. IAPR Workshop on Machine Vision Applications*, 2000 pp. 359-362.

Face Detection on Still Images Using HIT Maps*

Ginés García Mateos¹ and Cristina Vicente Chicote²

¹ Dept. Informática y Sistemas
University of Murcia, 30.170 Espinardo, Murcia, Spain
ginesgm@um.es

² Dept. Tecnologías de la Información y las Comunicaciones
University of Cartagena, 30.202 Cartagena, Murcia, Spain
cristina.vicente@upct.es

Abstract. We present a fully automatic solution to human face detection on still color images and to the closely related problems of face segmentation and location. Our method is based on the use of color and texture for searching skin-like regions in the images. This is accomplished with connected component analysis in adaptatively thresholded images. Multiple candidate regions appear, so determining whether each one corresponds or not to a face, solves the detection problem and allows a straightforward segmentation. Then, the main facial features are located using accumulative projections. We present some results on a database of typical TV and videoconference images. Finally, we extract some conclusions and advance our future work.

1 Introduction

Most of the existing techniques for face detection suffer from being either quite expensive or not very robust. In the first group, we can find systems that are based on exhaustive multiscale searching using neural networks [4] or eigen-decomposition [3] and, usually, color is not used. On the other hand, systems that make use of color features [5], [6], are computationally less expensive but are not very robust and present serious problems under uncontrolled environments.

The research described in this paper deals with the problem of human face detection on color images and the closely related problems of face segmentation and facial features location. We propose a technique based on color features which is intended to work under realistic uncontrolled situations. It has been tested using a database of images acquired from TV and from a webcam, achieving very promising results.

The key point in face analysis using color images is to search and describe skin-like regions [6]. We have defined a representation space named HIT (Hue, Intensity and Texture), that allows a simpler detection of skin-like regions. A fast connected component labeling algorithm is applied on thresholded HIT images, using adaptive thresholding in order to achieve invariance to clutter, noise and intensity.

* This work has been partially supported by Spanish CICYT project TIC1998-0559.

2 Skin-Region Searching in HIT Maps

2.1 HIT Maps

Differences among typical images of human skin (due to environment conditions, the acquiring system or the skin itself) do mainly cause intensity variations [7]. Thus, the color spaces most widely used for skin analysis are designed to be intensity invariant. We can mention the chromatic color space, or normalized (r, g) [7] and the HSV or HSI spaces [2],[5],[6], among others. But in many non-trivial cases, color features are not enough to separate skin from background objects. In these cases, intensity gradient is useful to detach face from other objects, and intensity itself may also be useful.

We have defined a representation space, named HIT, so that each RGB input image is transformed, in a preprocessing stage, into a three-channel image: *Hue*, *Intensity* and *Texture*. This *Texture* is defined as the magnitude of the multispectral gradient in RGB using the Sobel operator. The use of this particular color space transformation is justified in detail in [2]. Fig. 1 shows two sample images used in the tests, and their corresponding HIT transformations.

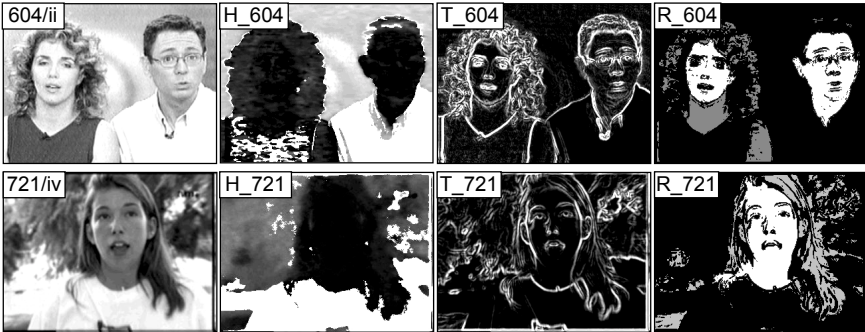


Fig. 1. Skin-region searching in HIT maps. From left to right: input image; hue channel; texture channel; skin regions found using adequate thresholds (those verifying size and shape criteria, in white).

2.2 Skin-Region Searching

A classification process is defined on HIT patterns, so that each vector $v = (h, i, t)$ is classified into one of two classes: skin or non-skin. We use a simple thresholding on the three channels, thus simplifying the training and classification. Contiguous pixels that are classified as skin patterns are then joined into skin-like regions, using a connected component labeling algorithm. This algorithm can be implemented with a single and very efficient scan of the image, from left to right and from top to bottom.

This method works quite well when adequate thresholds are selected, as in Fig. 1. But these thresholds may change from one image to another, so they can not be a priori fixed. Fig. 2 shows a sample of the variety that skin color and texture may undergo in different images, due to the environmental lighting conditions, the acquiring system and the noise in the video signal.

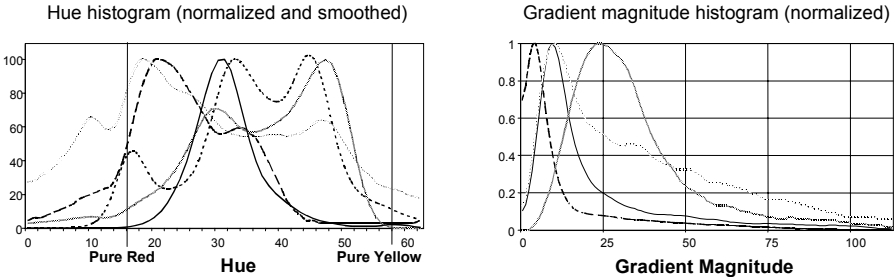


Fig. 2. Hue and Texture channel histograms for some skin-color regions in various images.

Our proposal is to use these histograms in order to adapt the color and texture models for each image in particular. For the color model, the hue of the skin corresponds to a maximum in that histogram, lying between pure red and pure yellow. For the texture and intensity channels, thresholds are calculated using the corresponding histograms. These thresholds can be more or less restrictive, so different adaptative tests appear. The whole process of skin-region searching is depicted in Fig. 3.

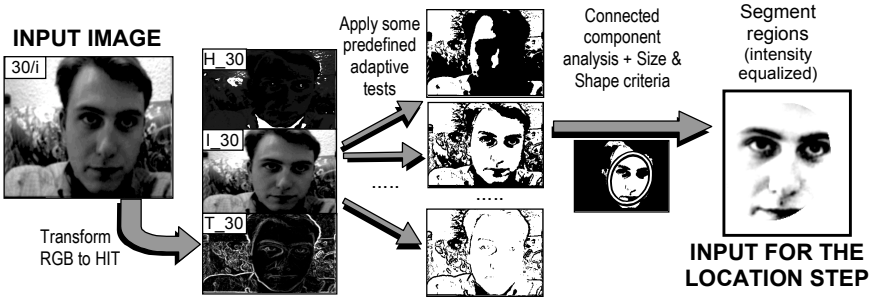


Fig. 3. Candidate skin-region searching process. The output are segmented candidate regions.

First the input image is transformed into an HIT map. Using the histograms, some adaptative tests are defined, that result in binarized images. In the experiments, six of these tests have been applied to each image. Connected components are then searched in binarized images. We apply shape criteria on the resulting regions to select those ones with elliptical shape that are not very elongated. These criteria are described in more detail in [2]. Finally, the regions that satisfy both criteria are segmented, equalizing the intensity channel with a linear model of the skin-region intensity, as in [4]. The segmented area corresponds to the ellipse that better fits the candidate region.

3 Facial Features Location

From the segmented regions obtained through the previous steps, the main facial features (i.e. eyebrows, eyes, nose and mouth) can be accurately located by analysing its horizontal and vertical integral projections. Besides, the a priori knowledge about the human facial structure, makes it possible to apply some heuristics in order to guide the location process in an efficient and smart way.

Integral projections on edge images and on intensity images [1], [6], have proved to be useful for facial features location. Given a segmented grayscale input image $I(x,y)$, its horizontal and vertical integral projections are defined as $HP(y) = \sum I(\bullet, y)$ and $VP(x) = \sum I(x, \bullet)$. These projections are smoothed in order to remove some small spurious peaks. Gaussian masks of different size are used for this purpose, depending on the size and contrast of the input image. Then, the location of the facial features can be obtained from the local maxima and minima extracted from these softened projections, as shown in Fig. 4. If no prominent peaks are found in the expected positions, then we infer that there is no face in the image. This detection test implies that a face exists when all its features are located. This way, the number of false-positive errors (see Table 1) is very small. But many existing faces are difficult to be located, so we can also consider that a face is detected when a candidate region is found. We will denote these two possibilities as detection *before location* and *after location*.

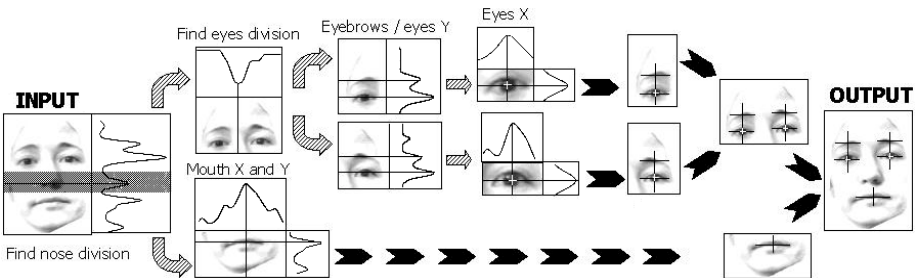


Fig. 4. Facial components location. The segmented region is successively divided into smaller ones by finding the maxima and/or minima within their softened integral projections and by applying some a priori knowledge about the face geometry.

4 Experimental Results

For our experiments, we have constructed a database of color images containing human faces within complex environments. Most of the existing face databases used for face detection are composed by gray scale images [4]. On the other hand, color face databases, used for person recognition, are not adequate for detection benchmarks. At the moment, our database is composed by 195 color images, some of them not corresponding to faces or containing more than one. A total of 101 distinct individuals appear on them, out of 199 existing faces. All these images have been acquired from a webcam and from 27 different TV channels, a few of them rather noisy.

The images have been classified into six groups: i) videoconference images; ii) news with presenters' faces and shoulders only; iii) news with presenters' faces and busts; iv) TV series, documentaries and outsides; v) fashion shows; and vi) non-faces. The detection and location results achieved by our system, are shown below.

Table 1. Face detection and location results.

Image group	Existing faces (images)	Faces detected before / after location	False-positive before / after location	False-negative before / after location	Location accuracy error
i)	35 (35)	34 / 30 97.1% / 85.7%	0 / 0 0% / 0%	1 / 5 2.9% / 14.3%	4.1 %
ii)	14 (14)	13 / 9 92.9% / 64.3%	1 / 0 7.1% / 0%	1 / 5 7.1% / 35.7%	3.6 %
iii)	58 (55)	53 / 35 91.4% / 60.3%	21 / 0 38.2% / 0%	5 / 23 8.6% / 39.7%	2.4 %
iv)	78 (69)	63 / 30 80.8% / 38.5%	27 / 4 39.1% / 5.8%	15 / 48 19.2% / 61.5%	1.9 %
v)	14 (13)	11 / 10 78.6% / 71.4%	3 / 0 23.1% / 0%	3 / 4 21.4% / 28.6%	6.2 %
vi)	0 (9)	0 / 0 - / -	2 / 0 22.2% / 0%	0 / 0 - / -	-
TOTAL	199 (195)	174 / 114 87.4% / 57.3%	54 / 4 27.7% / 2.1%	25 / 85 12.6% / 42.7%	3.1 %

A comparison of some state-of-the-art methods related to face detection can be found in [4], where the detection rates exhibited are between 78.9% and 90.5%. Compared to them, our method achieves similar results but requires less expensive computations. The performance highly varies from the most favorable group, that of videoconference images, with 85.7% of the faces located, to the more difficult one, group iv), with only 38.4% of the faces located. However, the false-positive rate *after location* is always very low, with a total of 2.1%.

The location results have been compared with a manual measure of the facial features positions. The location accuracy with respect to the face height, is nearly always below 4%, with a mean accuracy error of 3.1%. Some results of our detection method are shown in Fig. 5. Note that a faced is said to be detected *before location* when a box appear, and *after location* when all the facial features (marked with +) are found.

5 Conclusions and Future Work

The method here described, offers a unified solution to three basic problems of face analysis: detection, segmentation and location. We have defined the HIT representation space, which takes into account color, intensity and texture information. Adaptive techniques are used for establishing color and texture models of the images. Then, the main facial features are located by searching certain local maxima in the integral projections of intensity images, given some a priori geometrical constraints.

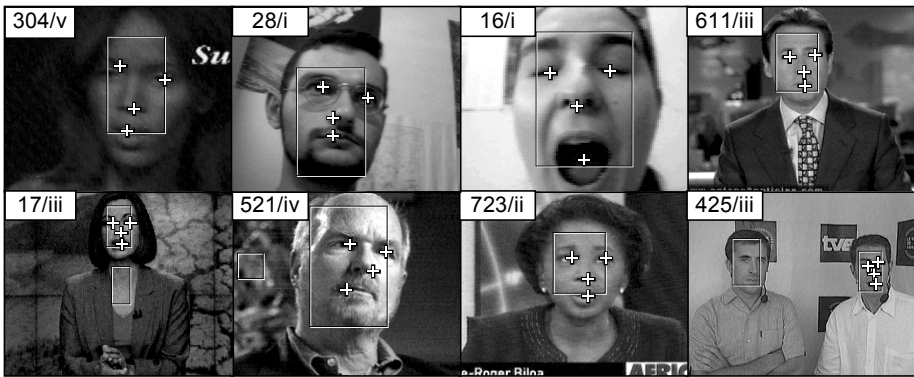


Fig. 5. Face detection results in some sample images. Candidate regions are indicated with a bounding box. Eyes, nose and mouth are marked if they have been located.

The achieved results are very promising, with a total detection ratio *before location* of 87.4%. This performance decreases when the location step is considered. This is due to a bad segmentation, which includes hair or neck as a part of the face region. We are currently working on improving segmentation, but it is worth to note the very low false-positive error obtained (2.1%) with respect to the number of images.

Our future work also includes the application of our method to person identification, facial expression recognition, videoconference coding and the extension of our approach to face tracking in image sequences.

References

1. Brunelli, R. and Poggio, T.: Face Recognition: Features versus Templates. IEEE Transactions on PRA, Vol. 15, No. 10 (October 1993) 1042-1052.
2. Garc a, G. and Vicente, C.: A Unified Approach to Face Detection, Segmentation and Location Using HIT Maps. SNRFAI2001, Castell n de la Plana, Spain, (May 2001).
3. Moghaddam, B. and Pentland, A.: Probabilistic Visual Learning for Object Detection. International Conference on Computer Vision, Cambridge, MA (1995).
4. Rowley, H.A., Baluja, S., and Kanade, T.: Neural Network-Based Face Detection. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 20, No. 1 (January 1998) 23-38.
5. Sigal, L. et al.: Estimation and Prediction of Evolving Color Distributions for Skin Segmentation Under Varying Illumination. IEEE Conference on CVPR (2000).
6. Sobottka, K. and Pitas, I.: Looking for Faces and Facial Features in Color Images. PRIA: Advances in Mathematical Theory and Applications, Vol. 7, No. 1 (1997).
7. Yang, J. and Waibel, A.: A Real-Time Face Tracker. In Proceedings of the Third IEEE Workshop on Applications of Computer Vision (1996) 142-147.

Lip Recognition Using Morphological Pattern Spectrum

Makoto Omata, Takayuki Hamamoto, and Seiichiro Hangai

Department of Electrical Engineering, Science University of Tokyo
omata@hanlab.ee.kagu.sut.ac.jp
{hamamoto, hangai}@ee.kagu.sut.ac.jp

Abstract. For the purpose of recognizing individuals, we suggest a lip recognition method using shape similarity when vowels are uttered. In the method, we apply the mathematical morphology, in which three kinds of structuring elements such as square, vertical line and horizontal line are used for deriving pattern spectrum. The shapeness vector is compared with the reference vector to recognize individual from lip shape. According to experimental results with 8 lips uttered five vowels, it is found that the method successfully recognizes lips with 100% accuracy.

1 Introduction

A study of biometrics technology is increasing lately, in order to prevent criminals from entering buildings and/or strengthen security for various applications. Generally, fingerprints, voices, faces, irises, retinas, palms and signatures are known as biometrics-features [1]. Among these, facial information includes many familiar features such as shape of eyebrow, shape and color of eye, shape of nose, shape of lip, position of parts and outline of face. Therefore, if we can extract respective features correctly and combine them, it is well expected to make a reliable identification or verification system.

From this point of view, various studies concerning face shape recognition have been performed [2,3]. Some of those, however, contain indefinite information such as eyeglasses, hairstyle, etc [4]. So, it is difficult to recognize only by facial shape.

By this reason, we suggest a lip recognition method using morphological pattern spectrum. In the method, we use three kinds of structuring elements, i.e., square, vertical line and horizontal line. The shapeness vector, calculated from the pattern spectrum, is compared with reference patterns indexed by uttered vowel information, and the individuals are determined.

In this paper, overview of the system, pattern spectrum used in this study and experimental results are given.

2 Overview of the Recognition System [5]

Figure 1 shows the overview of the lip recognition system. Details of each process are as follows.

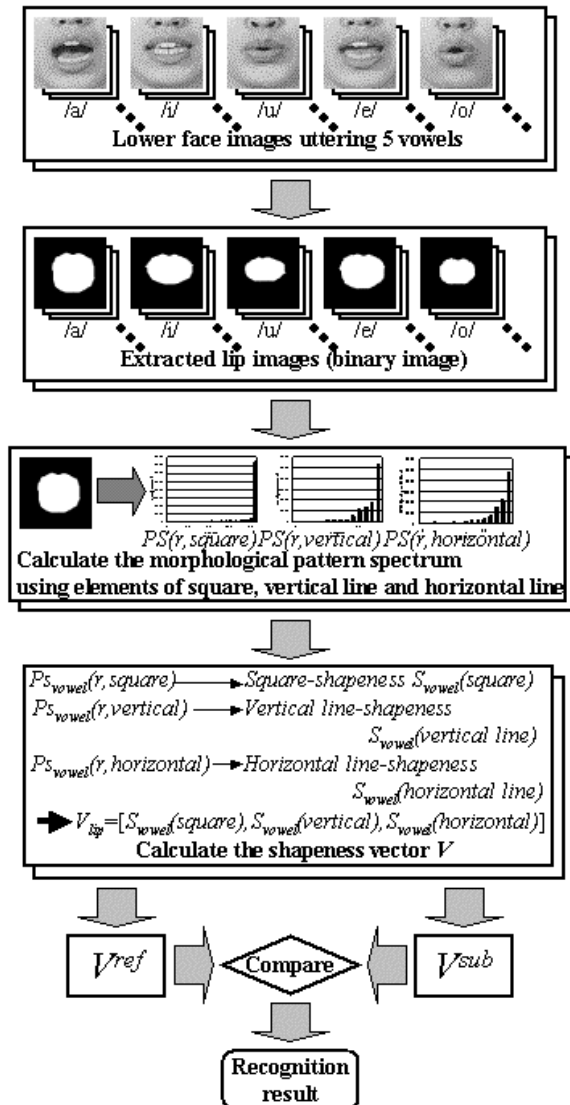


Fig. 1. Overview of the Lip Recognition System.

2.1 Lip Area Extraction

The lip area is extracted by the following steps.

<Step1> Reddish Area Extraction with a^* Threshold

In the CIE1976 $L^*a^*b^*$ uniform color space, parameter " a^* " represents "reddish" and is suitable for extracting lip area. The threshold value is determined by comparing the average and the deviation of a^* in skin area and in lip area. a^* is obtained as follows, [6]

$$a^* = 500 \left\{ \left(\frac{X}{X_0} \right)^{\frac{1}{3}} - \left(\frac{Y}{Y_0} \right)^{\frac{1}{3}} \right\} \quad (1)$$

where X, X_0, Y, Y_0 , are as follows,[7]

$$\begin{aligned} X &= 0.607R + 0.174G + 0.200B \\ Y &= 0.299R + 0.587G + 0.114B \\ X_0 &= 98.072 \times 2.55, \quad Y_0 = 100.000 \times 2.55 \end{aligned} \quad (2)$$

<Step2> Islands Removal

It is well expected that the pixel which have the maximum value of a^* is a part of lip area. In order to remove ambiguous areas (islands) in the image after Step1, we leave only the area containing the pixel with maximum a^* .

<Step3> Template Matching

By using two kinds of lip templates, the upper lip area and the lower lip area are extracted. Simultaneously, unnecessary areas, which combine with the true lip area, are also removed. In matching, Sequential Similarity Detection Algorithm [8] is used.

2.2 Calculation of Morphological Pattern Spectrum [9]

Mathematical morphology consists of various operations. In this study, we use Minkowski addition, Mikowski subtraction and Opening for deriving pattern spectrum. Basic three operations are as follows, where X is a set targeted in operation, B is a set named Structuring element, and each set's element is x and b . B^s means a symmetric set of B .

Minkowski addition:

$$X \oplus B = \{z \in E : z = x + b, x \in X, b \in B\} \quad (3)$$

Minkowski subtraction:

$$X \ominus B = \{z \in E : z - b \in X, b^s \in B\} \quad (4)$$

Opening X_B :

$$X_B = (X \ominus B^s) \oplus B \quad (5)$$

Pattern spectrum shows that a structuring element (B) represents the position of a target area (X) as distribution of the shape of structuring element or Scale when the scale and the shape of structuring element are determined. The scale shows the size of structuring element. Pattern

spectrum $PS_X(r, B)$ is given by the following equation, where $A(X)$ means that the area of X and r is the scale. (r is greater than or equal 1)

$$PS_X(r, B) = - \frac{dA(X_{rB})}{dr} \quad (6)$$

Figure 2 shows pattern spectrum of lip image uttering vowel /a/ when 3x3 pixels square is used as a structuring element. The higher a spectra at the maximum r , the more alike an area X is as a structuring element B . Shapeness is a likeness between X and B . B-shapeness $S_X(B)$ is as follows, where r_{max} means the maximum of scale r .

$$S_X(B) = \frac{PS_X(r_{max}, B)}{A(X)} \quad (7)$$

$$S_X(B) = \frac{PS_X(r_{max}, B) + PS_X(r_{max} - 1, B)}{A(X)} \quad (8)$$

If X is completely similar to B , $S_X(B)$ equals to 1. Equation (8) is used when the spectra at r_{max} is not maximum.

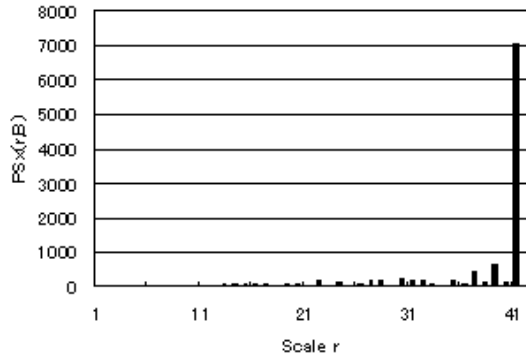


Fig. 2. An Example of Pattern Spectrum.

2.3 Feature Extraction and Recognition

Three structuring elements(B) are shown in Figure 3. The proposed lip recognition is done by the following two processes.

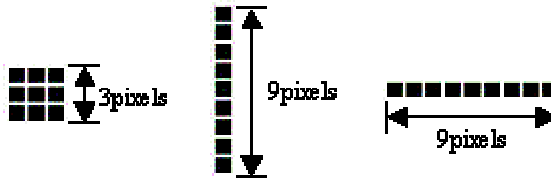


Fig. 3. Structuring elements. square(left), vertical line(center) and horizontal line(right).

2.3.1 Lip Recognition Process Using Euclidean Distance

- (i) Lip areas are extracted from each lip images by the method of section 2, and each lip areas are painted out inclusive inner lip as Figure 3 (right).
- (ii) Pattern spectrums are obtained from these binary lip shape images by using three structuring elements such as Figure 4 (square, vertical line and horizontal line).
- (iii) Square-shapeness is calculated from the pattern spectrum of square by equation (7). Two line-shapenesses are calculated from each line s pattern spectrum by equation (8). These shapenesses compose of a vector V .

$$V_{vowel} = \begin{bmatrix} S_{vowel}(square) \\ S_{vowel}(vertical\ line) \\ S_{vowel}(horizontal\ line) \end{bmatrix} \quad (9)$$

- (iv) Reference vector V^{ref} is calculated from one lip dataset. And subjective vector V^{sub} is calculated from other lip dataset. Each vowel s and each speaker s V^{sub} is compared with each vowel s and all speaker s V^{ref} . The speaker whose V^{ref} is the nearest to V^{sub} in Euclidean space is chosen as output;

$$\begin{array}{l} \text{Speaker} \quad \text{Output}_{vowel} \\ = \begin{cases} A \quad (|Speaker V_{vowel}^{sub} - A V_{vowel}^{ref}| \text{ is minimum}) \\ B \quad (|Speaker V_{vowel}^{sub} - B V_{vowel}^{ref}| \text{ is minimum}) \\ \vdots \\ H \quad (|Speaker V_{vowel}^{sub} - H V_{vowel}^{ref}| \text{ is minimum}) \end{cases} \end{array}$$

(A, B, C, D, E, F, G and H are the name of speakers.)

- (v) Each vowel s recognition result is decided by majority of these speaker s outputs.
- (vi) A final result is decided by majority of the each vowel s recognition results.

2.3.2 Lip Recognition Process Using Neural Networks

Process (i) (iii) are same as the section 2.3.1. (i)-(iii).

- (iv) Reference vector V^{ref} is calculated from one lip dataset. And subjective vector V^{sub} is calculated from other lip dataset. Each vowel s neural network (neural network s details are shown in Table 1) is made by training each vowel s and all speaker s V^{ref} . Each vowel s and each speaker s V^{sub} are fed through the neural networks.
- (v) Recognition based on vowels is done by the majority.
- (vi) Final recognition is also done by the majority of vowel based results.

Table 1. Details of the Neural Network.

The number of the unit at the input layer	3
The number of the unit at the hidden layer	40
The number of the unit at the output layer	8
Training time	400000
Training method	Back propagation method

3 Experimental Results

Table 2 shows a condition of data acquisition. Table 3 shows results of the recognition process by using Euclidean distance. In Table 3, the result of Vowels is the majority of each vowel's outputs, and the result of Final result is the majority of five vowel's recognition results. Result of speaker *B*'s /o/ and speaker *H*'s /u/ and /o/ each have two results because these each two vowels are same number of outputs.

This result shows that the method of this process can discriminate 8 Japanese male with 75.0% precision. It means that the shapeness vector has individual information of the speaker's lip shape. So we thought that recognizing by using neural networks would raise a success rate.

Table 4 shows results of the recognition process by using neural networks.

This result shows that the method of this process can discriminate 8 Japanese male with 100.0% precision. It shows that the neural network is effective for categorizing shapeness vectors.

Table 2. An Experimental Condition.

Speaker	8 (Japanese male)
Rec. data	2 sets of Japanese five vowels (/a/, /i/, /u/, /e/, /o/), 10 frames / speaker.
Rec. condition	Under fluorescent lights, Using DCR-VX9000 (SONY)

Table 3. Results of Lip Recognition (Euclidean Distance).

Speaker	<i>A</i>						<i>B</i>					
	Vowels					Final result	Vowels					Final result
	a	i	u	e	o		a	i	u	e	o	
Majority outputs	H	A	A	A	A	A	F	G	H	F	C	F
Speaker	<i>C</i>						<i>D</i>					
Majority outputs	C	C	C	A	C	C	G	D	G	A	G	G
	<i>E</i>						<i>F</i>					
Majority outputs	E	B	E	E	E	E	F	F	F	F	F	F
Speaker	<i>G</i>						<i>H</i>					
Majority outputs	E	G	G	G	G	G	C	B	F	H	F	H

Table 4. Results of Lip Recognition (Neural Networks).

Speaker	A						B					
	Vowels					Final	Vowels					Final
	a	i	u	e	o	result	a	i	u	e	o	result
Majority outputs	D	A	A	A	A	A	B	G	B	E	H	B
							F					
Speaker	C						D					
Majority outputs	C	C	C	C	C	C	D	D	D	D	D	D
Speaker	E						F					
Majority outputs	E	G	E	E	E	E	F	F	F	F	F	F
Speaker	G						H					
Majority outputs	G	G	G	G	A	G	H	G	F	H	H	H

4 Conclusion

In this paper, we have proposed the lip recognition method using morphological pattern spectrum. And, the effect of the neural networks was also compared. From the experimental results, it was shown that the shapeness vector V was available information for individual recognition by lip shape and 8 Japanese lips could be classified with 100.0% accuracy.

However, we think that the proposed method is not sophisticated yet. To improve the classification accuracy, we will consider a new structuring element, e.g. rectangle, ellipse or asymmetrical shape.

References

1. Y. Hayashi, Technique, Needs and expectation of Personal Identification , *Journal of the Society of Instrument and Control Engineers*, Vol.25, No.8, pp. 683-687, 1986.
2. T. Minami, Personal Identification of Face Images , *Systems, Control and Information*, Vol.35, No.7, pp. 415-422, 1991.
3. S. Akamatsu, Computer Recognition of Human Face A Survey- , *Transaction of IEICE D-II*, Vol.80-D-II, No.8, pp. 2031-2046, 1997.
4. Y. Saito and K. Kotani, Extraction and Removal of Eyeglasses Region in Facial Image using Parametric Model of Eyeglasses Frame Shape , *Technical Report of IEICE*, CS97-140, pp. 55-60, 1997.
5. M. Omata, H. Machida, T. Hamamoto, and S. Hangai, Extraction of Lip Shape for Recognizing Phonetics under Noisy Environment , *Proceedings of the 2000 IEICE General Conference*, D-12-65, 2000.
6. A. Kakisaki, M. Sugihara, T. Hanamura, and H. Tominaga, Image Coding Control Method based on Color Difference in Uniform Color Space , *Proceedings of the Picture Coding Symposium of Japan (PCSJ93)*, pp. 131-132, 1993.
7. Y. Shirai and M. Yachida, Pattern Information Processing , pp.105-107, *Ohmsha*, 1998.
8. H. Shibayama, Image Processing using X11 Basic Programming , pp. 250-256, *Gijutsuhyoronsya*, 1994.
9. H. Kobatake, Morphology , *Corona Publishing*, 1996.

A Face Location Algorithm Robust to Complex Lighting Conditions

Robert Mariani

RWCP * Multi-Modal Functions KRDL ** Laboratory
21 Heng Mui Keng Terrace, Kent Ridge, Singapore 119613
Tel: (+65) 874-8810 Fax: (+65) 774-4990 rmariani@krdl.org.sg

Abstract. In this paper, we present a face location system in a complex background and robust to a wide range of lighting conditions, likely to appear in an indoor environment. The complete system contains two parts: the face locator and the face tracker. We will only describe the face locator. Face hypothesis are obtained combining a pyramidal grey-level template matching and a geometrical measure based on the facial feature organization. Classification to face or non-face is realized by linear discriminant analysis (LDA) on the principal components analysis (PCA) of a 26-dimensional feature vector extracted from the face hypothesis. Experiments on 1500 images in a cluttered background with 12 lighting conditions are very encouraging.

1 Introduction

This work is included in a project for face recognition in a video sequence of images [1]. Existing techniques include template-matching [2], model-based [3], and color-based [4] approaches. Here, we propose a face location system robust to the lighting conditions in an indoor environment. The images in figure 1, of the same person, are extracted automatically by our system, and are the kind of situations we wish to handle. These lighting conditions make most of the available techniques insufficient, especially the color and contour-based techniques.



Fig. 1. Same person, different lights.

Our method combines a multi-scale grey-level template matching, especially designed to be robust to the lighting conditions, and a geometrical measure

* Real World Computing Partnership.

** Kent Ridge Digital Laboratory.

based on facial features organization, which approximates a model-based approach, without paying attention to the precision of the facial feature organization. Classification to face or non-face is realized by LDA on the PCA of a 26-dimensional feature vector extracted from the face hypothesis.

2 Fixed-Size Template Matching

We research in an image of 320x240 pixels, a face having a size varying from 30x50 pixels to 50x80 pixels. This research is realized at three different scales, namely in an image of 80x60, 160x120 and 320x240. For this, we use 3 face templates, having respectively a size of (40,60), (20,30) and (10,15) pixels.

Let I_1, I_2, I_3 the three input images, where I_1 is the full resolution image, and let T_1, T_2, T_3 the three corresponding face templates. We denote $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ the corresponding pixel (x, y) of the image I_1, I_2 and I_3 respectively, with the relation $(x_{n+1}, y_{n+1}) = (x_n/2, y_n/2)$, and $(x, y) = (x_1, y_1)$. For a given pixel (x, y) of the original image I_1 , we derive two measures, $M(x, y)$ and $P(x, y)$, obtained from the three correlation scores $\rho_1(x, y), \rho_2(x, y)$ and $\rho_3(x, y)$, between the template T_n centered at (x_n, y_n) and the image $I_n, n = 1..3$. The coefficient of correlation between X and Y is defined as $\rho(XY) = \frac{E(XY) - E(X)E(Y)}{\sigma(X)\sigma(Y)}$ where $E(X)$ is the mean and $\sigma(X)$ the standard deviation of X .

To deal with the lighting conditions that affect the correlation score, we have slightly modified the template matching. The extreme case being one side of the face very dark, and the other very bright, as seen in figure 1. This case illustrates a non-linear alteration of the grey-level of the same neutral face. As the statistical correlation is not invariant to non-linear shift, we compensate this drawback by splitting vertically the template and the observation, by computing the left-part correlation $\rho(X_l, Y_l)$ and the right-part correlation $\rho(X_r, Y_r)$ and by taking the maximum: $\rho(XY) = \max(\rho(X_l, Y_l), \rho(X_r, Y_r))$.

To speed-up the construction of the maps $M()$ and $P()$, we propose the following algorithm implemented in an efficient way: let ρ_1, ρ_2, ρ_3 the three correlation scores associated to the pixel (x, y) of the original image, and $\rho_1^*, \rho_2^*, \rho_3^*$ the maximum of correlation already computed from the image.

$$\begin{aligned}
 & \text{If } (\rho_1 > \alpha\rho_1^*) \wedge (\rho_2 > \alpha\rho_2^*) \wedge (\rho_3 > \alpha\rho_3^*) \\
 & \quad M(xy) = \max(\rho_1, \rho_2, \rho_3), P(xy) = \rho_1 * \rho_2 * \rho_3 \\
 & \quad \text{If } (\rho_1 > \rho_1^*) \rho_1^* = \rho_1 \\
 & \quad \text{If } (\rho_2 > \rho_2^*) \rho_2^* = \rho_2 \\
 & \quad \text{If } (\rho_3 > \rho_3^*) \rho_3^* = \rho_3
 \end{aligned}$$

At the beginning of the process, the maximum correlation scores $\rho_1^*, \rho_2^*, \rho_3^*$ are initialized to a minimum correlation score, 0.1. The constant α allows to consider the pixels having a relatively strong correlation ρ_1, ρ_2, ρ_3 compare to the best one $\rho_1^*, \rho_2^*, \rho_3^*$. At the end of the process, the map $P()$ is thresholded, $P(x, y) = 0$ if $\rho_1 * \rho_2 * \rho_3 < \alpha^3 \rho_1^* * \rho_2^* * \rho_3^*$, and the extraction of the local maxima of $P(x, y)$, in a window of (15,15) pixels, provides the K-best hypothesis of face

images having a size varying from (30,50) to (50,80) pixels. $M(x, y)$ will be used in the face classification step.

3 Geometrical Validation

At the previous step, we obtained the K-best face image hypothesis, each image having a size of (40,60) pixels. Here, we propose a first step to discard the non-faces while keeping all the faces, reducing the face hypothesis set from K to a maximum of 10 samples. We construct the map of all the equilateral triangles we can build from a binary version of the face, and if the maximum number of overlapping triangles around the center of the image (the face center hypothesis) is lower than a given threshold, then the face hypothesis is rejected.

The construction of the binary face image realizes a facial feature segmentation, when the observed image is actually a face. Again, we work on each vertical half of the image separately. For a given half-image, say the left half L, we proceed as follow: let (μ, σ) the mean and standard deviation of the grey-level of L, and C, the proportion of pixels p that verifies $|v(p) - \mu| \leq \sigma$ where $v(p)$ is the grey-level of p. The face hypothesis is rejected if σ is too low (< 5) or if C is too small ($< 50\%$). Otherwise, we associate to each pixel p of L a value $w(p)$, defined as $w(p) = 1$ if $v(p) \geq \mu$ or $w(p) = \frac{v(p) - \mu}{\sigma}$ if $v(p) \leq \mu$. Let w^* the threshold such that 15% of pixels p of L verify $w(p) < w^*$. The binary image is constructed by setting to 1 all the pixels that verify this property and 0 the others (cf. fig 2).

We consider the sets of connected pixels, and we discard the large connected components, usually corresponding to the hair when the observed image is a face. When the input image is a face, we expect at least 3 horizontal components, 2 eyes/eyebrows, 1 nose/mouth. After removing the large vertical components, the face hypothesis is rejected when the remaining components are less than 3.

We have derived a measure that discards a lot of false hypothesis, while keeping the correct face hypothesis. It is based on the construction of equilateral triangles linking the pixels with value 1. Given 3 pixels a,b,c, belonging to three different connected components, where c is the lowest point, the triangle (a,b,c) is accepted if

$$\begin{aligned} \theta(ab) &< \frac{\pi}{4} \\ 5 &< \|\vec{ab}\|, \|\vec{bc}\|, \|\vec{ac}\| < 15 \\ \frac{\max(\|\vec{ab}\|, \|\vec{bc}\|, \|\vec{ac}\|)}{\min(\|\vec{ab}\|, \|\vec{bc}\|, \|\vec{ac}\|)} &< 2 \\ \frac{\pi}{6} &< \widehat{cab}, \widehat{abc}, \widehat{bca} < \frac{\pi}{2} \end{aligned}$$

$\theta(x)$ is the angle of the vector \vec{x} with the horizontal, in absolute value, $\|\vec{x}\|$ denotes the norm of the vector \vec{x} , and \widehat{xyz} the angle at the point y. By analogy with the hough transform, we start with an accumulator initialized to 0 everywhere. For each valid triangle explored, we increment by 1, the value of all the cells falling inside this triangle. At the end, each cell of the accumulator indicates the number of triangle overlapping at its corresponding point. When the observed image is a face, we expect a large value around the face center. To

be more robust, we cumulate the values of the accumulator in a small window around the center of the face. Then, if the accumulated score is too low (< 50), the face hypothesis is rejected.

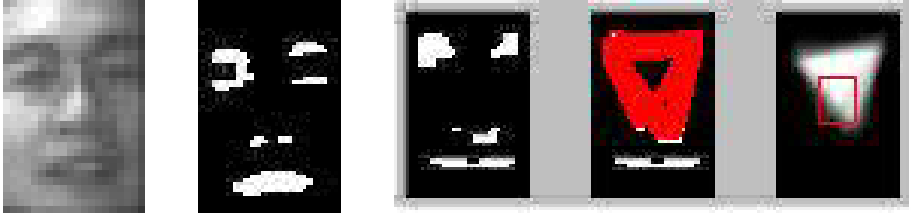


Fig. 2. Binarization and accumulation.

4 Classification Face/Non-face

A face hypothesis, representing a portion of the original image centered at (x,y) , is described by a 26 dimensional feature vector X , constructed using the contrast-normalized higher-order local autocorrelations proposed by Otsu [5] (24 components), the maximum of correlation $M(x,y)$ and the number of triangle $T(x,y)$. The feature vector X is then fed to a classifier to accept or reject the face hypothesis. For this, we designed 2 classifiers, namely a 'face'-classifier and a 'non-face'-classifier, and then, by combining their response, we decide the final class of X .

The optimal face classifier and non-face classifier have been found separately by supervised training, using the sequential backward algorithm for feature selection, minimizing at each level of the recursion, the classification error.

To obtain the face-classifier, we proceed as follow. Let F_N and G_N the face and non-face sets, where the feature vectors have N components, 26 at the beginning: 1) PCA : we extract the K leading eigenvectors of the covariance matrix of F_N ; 2) LDA : we estimate the optimal linear discriminant function that separates the projection of F_N and G_N , onto the subspace spanned by these K vectors, and this for $K = 1..N$. This process is repeated by considering the N couples $\{(F_{N-1}^1, G_{N-1}^1), (F_{N-1}^2, G_{N-1}^2), \dots, (F_{N-1}^N, G_{N-1}^N)\}$, where F_{N-1}^n and G_{N-1}^n are the face and non-face subsets of F_N and G_N , where the n th component of the feature vectors have been removed, leading to feature vectors of dimension $N-1$. If the classification error has not decreased, we keep the PCA-LDA face-classifier obtained with F_N and G_N . The non-face classifier has been obtained using the same method. A description of PCA and LDA techniques can be found in [5].

We propose a decision rule, based on the statistical distribution of the projected data. Let A and B the face and non-face training sets, $f()$ and $g()$ the discriminant functions for each class. Let $FA = f(A)$, $GA = g(A)$, $FB = f(B)$,

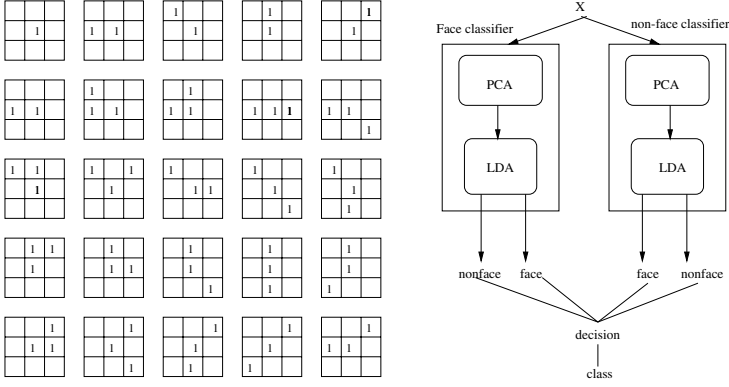


Fig. 3. (a) Autocorrelations masks; (b) face/non-face classifier.

$GB = g(B)$ the 4 1-dimensional clusters, images of A and B by $f()$ and $g()$. We derive $(\theta_{fa} \cap \sigma_{fa}) \cap (\theta_{fb} \cap \sigma_{fb}) \cap (\theta_{ga} \cap \sigma_{ga}) \cap (\theta_{gb} \cap \sigma_{gb})$ the mean and standard deviation of $FA \cap FB \cap GA \cap GB$ respectively. Given an input vector x , we obtain its 2 projections: $y_f = f(x)$ and $y_g = g(x)$. Let $y_{fa} = \frac{\clubsuit_f \otimes m_{fa} \spadesuit_f}{\sigma_{fa}}$, $y_{fb} = \frac{\clubsuit_f \otimes m_{fb} \spadesuit_f}{\sigma_{fb}}$, $y_{ga} = \frac{\clubsuit_g \otimes m_{ga} \spadesuit_g}{\sigma_{ga}}$, $y_{gb} = \frac{\clubsuit_g \otimes m_{gb} \spadesuit_g}{\sigma_{gb}}$, x is classified to the class of A or B according to the following sequential algorithm, where RA and RB are boolean standing for [reject A] and [reject B], and $label$ is the output class identifier:

If $\min(y_{fa} \cap y_{ga}) < \min(y_{fb} \cap y_{gb})$ $label=A$; else $label=B$
 $RA = RB = 0$
 If $(y_{fa} > 3\sigma_{fa}) \cap (y_{ga} > 3\sigma_{ga})$ $RA = 1$
 If $(y_{fb} > 3\sigma_{fb}) \cap (y_{gb} > 3\sigma_{gb})$ $RB = 1$
 If $(RA = 1) \cap (RB = 1)$ $label = B$
 If $(RA = 1) \cap (RB = 0)$ $label = B$
 If $(RA = 0) \cap (RB = 1)$ $label = A$

We obtained roughly 92% of correct classification for both classes, containing roughly 1000 faces and 10000 non-faces. This score represented, in this particular case, a great improvement compared to direct use of the Euclidean distance in the simplest two-classes classification problem, where we compute only one linear discriminant function.

We generalize this algorithm in order to find faces of different sizes, not restricted to a size varying from (30,50) to (50,80) pixels. This is realized by scaling the complete input image at 8 different resolutions, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2, and applying the previous algorithm to find the faces at this particular size. At the end of the face location, we obtain multiple face hypothesis. These faces are ranked using the product of $M(x,y) \cdot T(x,y)$, the greater the better. A final pruning algorithm is applied, starting from the best face hypothesis, removing all the face hypothesis that are spatially overlapping this face in the image, and

repeating this process for the remaining face hypothesis. The final result is the list of segmented faces in the image.

5 Results and Performance

We collected a database of 1500 images, of 15 persons in 12 lighting conditions, and with faces oriented in multiple direction, where color-based and contour-based techniques perform poorly. In 90% of cases, the real face was in the first position, in 93% of cases, in the 2 best positions, and in 98% of cases in the top five. The rejection of valid hypothesis was mainly due to the large variation of pose.

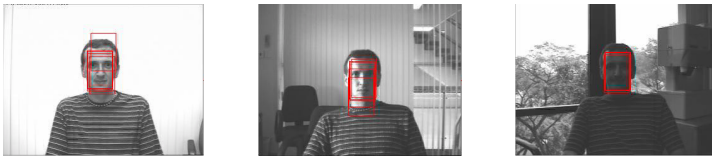


Fig. 4. Face location in complex lighting conditions.

6 Conclusion

We presented a multi-scale face location, robust to the lighting conditions, our main focus of research. Detection is based on a multi-resolution template matching, combined with a hough transform-like operation using a binary representation of the facial features. Classification is realized using two PCA-LDA classifiers and merging their responses for a final decision. Detection of faces at various sizes is handled by applying this algorithm at different resolution of the input image.

References

1. Huang, W. and Mariani, R., (2000). Face Detection and Precise Eyes Location. In *Proc 15th ICPR*, pp. 722-727, Barcelona, Spain, Sept. 3-7 2000.
2. Gutta, S. et al. (1996). Face Recognition Using Ensembles of Network. In *Proc 13th ICPR*, Vienna, Austria, Aug. 1996.
3. Yow, K.C and Cipolla, R. (1996). Feature-based human face detection. In *Tech. Report CUED/F-INFENG/TR 249*, Dept. of Engineering, Univ. of Cambridge, England, Aug. 1996.
4. Sun, QB, Huang, W., and Wu, JK., Face Detection Based on Color and Local Symmetry Information. In *Proc 3rd FG*, pp. 130-135, Nara, Japan, April, 14-16, 1998.
5. Hotta, K., Kurita, T., and Mishima, T., Scale Invariant Face Detection Method using Higher-Order Local Autocorrelation Features Extracted from Log-Polar Image. In *Proc 3rd FG*, pp. 130-135, Nara, Japan, April, 14-16, 1998.

Automatic Facial Feature Extraction and Facial Expression Recognition

S  verine Dubuisson, Fran  ck Davoine, and Jean-Pierre Cocquerez

Universit   de Technologie de Compi  gne, BP 20529, 60205 Compi  gne, France
{sdubui, fdavoine, cocquerez}@hds.utc.fr

Abstract. In this paper, we present an automatic algorithm for facial expression recognition. We first propose a method for automatic facial feature extraction, based on the analysis of outputs of local Gabor filters. Such analysis is done using a spatial adaptive triangulation of the magnitude of the filtered images. Then, we propose a classification procedure for facial expression recognition, considering the internal part of registered still faces. Principal Component Analysis allows to represent faces in a low-dimensional space, defined by basis functions that are adapted to training sets of facial expressions. We show how to select the best basis functions for facial expression recognition, providing a good linear discrimination: results prove the robustness of the recognition method.

1 Introduction

Face analysis can be divided into several subgroups: face detection, facial feature extraction, and recognition (such as face, age, gender, race, pose and facial expression recognition). The problem of facial expression recognition has recently emerged. In a general way, an expression is a combination of the Action Units (AUs) defined by Ekman and Friesen [4]: a local facial motion resulting from the compression or relaxing of facial muscles. A facial expression can be seen in two different ways: a motion in the face which requires working with video sequences and face motion analysis tools, or the shape and texture of the face, using statistical analysis tools, local filtering or facial feature measurement. The facial motion has been the first way to be explored by the researchers: a facial expression can be described by a global facial motion model ([1]), or by a set of local facial motion models ([11]) that researchers analyze along a video sequence by tracking facial feature points. In statistical analysis, we first have to learn what we are looking for: a learning method of the appearance of facial expressions using an eigenvector decomposition of the image space has been proposed in [9], which build a new representation subspace where a likelihood measure is computed to analyze new facial expression images. Local filtering methods change the definition domain of the images, which can help in feature extraction: researchers [7, 2] have used Gabor wavelets to reveal particular parts of the faces, which vary from a facial expression to another one. The amplitude of the Gabor wavelet response gives information about the facial expression.

2 Facial Feature Extraction

Most of a facial expression information is concentrated around facial features such as eyes or mouth: including irrelevant parts (hair, background, ...) can generate incorrect decisions for expression recognition. That is the reason why their detection is an important step before analyzing a facial expression. We briefly propose in this section a new algorithm for facial feature extraction, which is more precisely described in [3]. The position, scale and orientation of the face are detected using a generalized likelihood ratio-based detection approach described in [6]. Such algorithm returns a quadrilateral around the face and allows to compensate the global motion of the face. Thus, in the next steps, we will consider that we work with fixed size frontal-view face images.

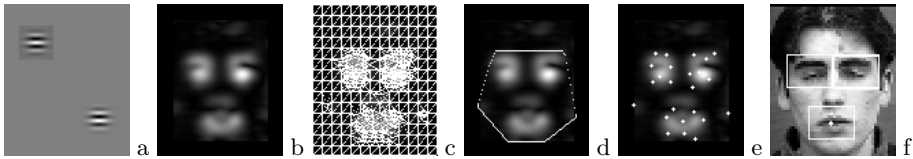


Fig. 1. Classification of the triangulation vertices: (a) Imaginary and real part of the Gabor kernel. (b) Magnitude of the filtered image. (c) Triangulation of (b). (d) Convex envelope of the triangulation. (e) First classification. (f) Detection of the three boxes containing the facial features.

Gabor Filtering. Images are then filtered using a Gabor kernel defined for a particular orientation and resolution (because of the constant size of the face images, we have an idea of the proportion of the features). Gabor wavelets best react to textures which are perpendicular to the orientation of their oscillation. We want to find the two eyes and the mouth in the face, that is the reason why we choose a Gabor kernel with an orientation of $\frac{\pi}{2}$.

Delaunay Triangulation and Clustering. Image content adaptive partitioning methods have shown their interest for image analysis, generating a lot of blocks on highly textured regions and, reciprocally, few larger blocks on less textured parts. We partition the magnitude of the filtered Gabor face images using a Delaunay triangulation. Finally, we get clouds of triangle vertices around feature regions of the Gabor filtered image (Fig. 1c) and these vertices are then clustered based on their position in the image to facilitate the localization of facial features using a K-means algorithm. Figure 1 illustrates the first steps of the clustering of the triangulation vertices: 378 vertices have been clustered into 5 different classes among which we can find the ones corresponding to the eyes and the mouth.

Graph Modeling. Starting with the nodes corresponding to the center of the previously extracted classes, we create a fully connected non-oriented graph in which we are looking for a particular triangle corresponding to the configuration

“left eye - right eye - mouth” Each node is labeled with its relative position in the graph and the average Gabor response in a fixed size centered box. Edges are labeled with their length. The triangle can be isolated from other by taking into account the geometrical facial feature constraints: we find the configuration of nodes which has large responses in a small neighborhood under these geometrical constraints.

Facial Feature Detection and Mask Extraction. We have an approximate idea of the facial feature positions and will localize them more precisely. We first search the best spatial configuration between the two boxes centered on the eye vertices (see previous part) using a Block-Matching algorithm (i. e. we try to minimize the root mean square error between the two boxes). The final position of the eyes in the face is obtained using both the Gabor and gradient response in the image. Once the eyes have been found, we look for the mouth using a sliding window moving along the median axis between the eyes, until both high magnitude values and good boundaries are found. Based on the boxes around the eyes and the mouth, we build a mask of the face to recover the nose and a great part of the cheeks (wrinkle can give an important information for facial expression analysis). Figure 2 shows some examples of facial feature detection (top) and facial masks extraction (bottom).

Results. We have tested this algorithm on 120 JPEG-compressed images from the FERET database [5] and on 550 images from the CMU-Pittsburgh database [6]: facial feature detection was respectively successful for 96% and 92% of these images. False detections were more often due to glasses or occlusion problems. Figure 2 gives some results of these detections: we can see that the algorithm is not perturbed by beards.



Fig. 2. Top: Facial feature extraction examples (FERET database) and illustration of the facial mask construction. Bottom: Examples of facial masks.

3 Facial Expression Analysis

3.1 Algorithm

The algorithm is divided into three main steps: Principal Component Analysis (PCA), search of the best basis vectors and Linear Discriminant Analysis (LDA).

A training set S is built using N mask images (previously normalized facial masks of size 68×72 : 4896-pixel vectors) belonging to c classes ($\frac{N}{c}$ images in each class). The usual method for classification (7) consist in first performing a PCA to reduce the dimensionality of the training set S : the eigenvalues λ_i of the total scatter matrix C are ranked in decreasing order and the global quality of the representation of the initial set is given by the inertia ratio $r_M = \frac{\lambda_1 + \dots + \lambda_M}{\text{trace}(C)}$. The most expressive vectors derived from a PCA process are those corresponding to the leading largest eigenvalues: M principal axes are selected and used to derive M -dimensional feature vector for each sample. These transformed samples are then used to execute LDA: we determine the mapping which simultaneously maximizes the between-class scatter S_B while minimizing within-class scatter S_W of the projected samples, so that the classes are separated as best as possible. The way to find the required mapping is to maximize the quantity $\text{trace}(S_W^{-1} S_B)$.

PCA and Best Base. Unfortunately, it is not guaranteed that the principal components corresponding to the largest eigenvalues define an optimal basis for the recognition problem. Thus, we consider an iterative procedure aiming at minimizing a generalization error of the classification system using the LDA process. PCA on the training data S has given a set of N principal components among which we are seeking the K “best” ones. To determine this best projecting base, we use a *step by step selection method*: during the step 1, we seek the “best” principal component among the N available. During the step j , we seek the principal component (among the $N - j + 1$ remaining) which, when added to those previously kept, is the “best” one. At each iteration j ($j = 1, \dots, N$), we successively select the available principal components and add them to those previously kept to build a projective subspace: if y_i^k denotes the j -dimensional feature vector extracted from the i^{th} sample (i.e. i^{th} projected sample) of the k^{th} class, let g_k ($k = 1, \dots, c$) be the mean vector of the k^{th} class and $G = \frac{1}{c} \sum_{k=1}^c g_k$ the total mean vector in this projective feature space. The within-class and between-class scatter matrix in this feature space can respectively be calculated as follows:

$$S_W = \sum_{k=1}^c \sum_{i=1}^{N/c} (y_i^k - g_k)^t (y_i^k - g_k) \quad \text{and} \quad S_B = \sum_{k=1}^c (g_k - G)^t (g_k - G)$$

We seek the principal component which maximizes $\text{trace}(S_W^{-1} S_B)$ (criterion of LDA) in the j -dimensional projective subspace. This process is repeated until all the available components are ranked. The selection criterion used to define the quality of a set of components is the linear discriminator generalization error. This error is estimated using a cross validation technique in which 80% of the data are used for learning and 20% for testing. The error is computed by randomly choosing the learning set and the test set several times and averaging the results. This algorithm allows to determine the optimal number K of principal components in decreasing order of importance for which the generalization error is minimum (e_{\min}). Figure 8 illustrates the case of the binary classifier “Sadness/Happiness”. These components define the subspace in which we will project our images. Figure 9 shows that the two classes of a set are well separated if it

is projected into the subspace constructed with the two “best” principal components rather than the one constructed with the two first principal components

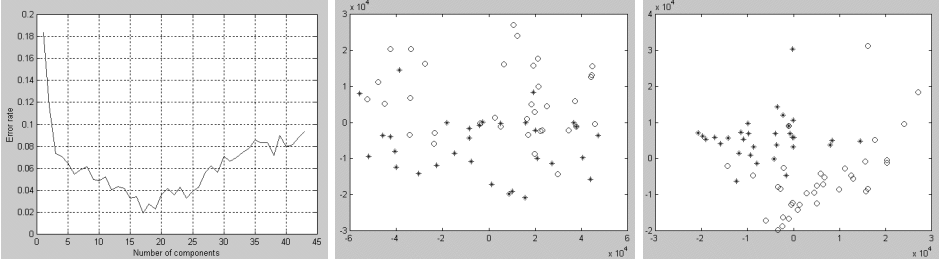


Fig. 3. From left to right: Generalization error rate for the binary classifier “Sadness/Happiness”: minimum classification error $e_{min} = 0.019$, $K = 17$, and the projection of these two classes on the PCA-subspace constructed with the two first principal components and the two “best” principal components.

3.2 Results

We have tested our method on images from the CMU-Pittsburgh [i](#) database. We construct a learning set using the masks obtained during the feature extraction phase (see Sect. [2](#)). We have trained 15 binary classifiers (one for each pair of facial expressions), using $N = 140$ mask images, and a 6-expression classifier using $N = 420$ mask images. Test images (which have not been learned) are introduced, projected into the PCA-subspace, and then into the LDA-subspace. A facial expression class is tested with the 5 binary classifiers in which it appears and an average of these results is computed to obtain the global percentage of correct classification, or directly recognized using the 6-expression classifier. We have compared the percentage of correct classification using three different space projections: S_1 is constructed with the first M sorted eigenvectors (M is the rank of the eigenvalue matrix for which the inertia is up to 90 %), S_2 with the K first eigenvectors and S_3 with the K “best” eigenvectors, given by the K “best” principal components (see section. [3.1](#)). In table [1](#) we compare the classification error rate of unknown images using binary and 6-expression classifiers. These tests show that the percentage of correct classification is improved after a projection into S_3 , and also that K eigenvectors are sufficient to define a subspace.

4 Conclusion

We have presented a new algorithm for automatic facial feature extraction and an improved method for facial expression analysis. The automatic facial feature extraction allows to isolate the most important part of a face for its expression

Table 1. Comparison of correct classification results after projection into three different subspaces, for binary and 6-expression classifiers.

	Exp. #	Surprise 81	Anger 41	Sadness 80	Happiness 83	Fear 53	Disgust 35	Total 373
Binary Classifiers	S1	92%	83%	93%	90%	90%	87%	89%
	S2	90%	84%	92%	89%	86%	88%	88%
	S3	96%	88%	95%	94%	90%	89%	92%
6-expression Classifier	S1	73%	71%	72%	73%	74%	63%	71%
	S2	70%	66%	68%	68%	74%	60%	68%
	S3	74%	73%	80%	77%	74%	72%	75%

analysis by extracting a facial mask. Then, the facial expression analysis is improved first because we only take into account the pixels in the facial mask, second because we optimize the reduction of the information by projecting into the best subspace for classification. We have shown how results are improved using this method.

References

- [1] J. N. Bassili. Emotion recognition: The role of facial movement and the relative importance of upper and lower areas of the face. *J. Personality and Social Psychology*, 37:2049–2059, 1979.
- [2] G. Donato, M. Stewart Barlett, J. C. Hager, P. Ekman, and T. J. Sejnowski. Classifying facial actions. *IEEE : Transactions on Pattern Analysis and Machine Intelligence*, 21(10):974–989, Oct. 1999.
- [3] S. Dubuisson, F. Davoine, and M. Masson. A solution for facial expression representation and recognition. In *ICAV3D*, Grece, May 2001.
- [4] P. Ekman and W. Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Calif.: Consulting Psychologists Press, 1978.
- [5] T. Kanade, J. F. Cohn, and Y. Tian. Comprehensive database for facial expression analysis. In *Proceeding of the Fourth International Conference of Face and Gesture Recognition*, pages 46–53, Grenoble, France, 2000.
- [6] C. Kervrann, F. Davoine, P. Perez, R. Forchheimer, and C. Labit. Generalized likelihood ratio-based face detection and extraction of mouth features. *Pattern Recognition Letters*, 18:899–912, 1997.
- [7] M. J. Lyons, J. Budynek, and S. Akamatsu. Automatic classification of single facial images. *IEEE : Transactions on Pattern Analysis and Machine Intelligence*, 21(12):1357–1362, Dec. 1999.
- [8] S.R. Rizvi, P.J. Phillips, and H. Moon. The feret verification testing protocol for face recognition algorithms. In *Proceeding of the Third International Conference of Face and Gesture Recognition*, pages 48–55, Nara, Japan, 1998.
- [9] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
- [10] Y. Yacoob and L.S. Davis. Recognizing human facial expressions from long image sequences using optical flow. *IEEE : Transactions on Pattern Analysis and Machine Intelligence*, 18(6):636–642, jun 1996.

Fusion of Audio-Visual Information for Integrated Speech Processing

Satoshi Nakamura

ATR Spoken Language Translation Research Laboratories
2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto, Japan

nakamura@slt.atr.co.jp

<http://www.slt.atr.co.jp/~nakamura>

Abstract. This paper describes the integration of audio and visual speech information for robust adaptive speech processing. Since both audio speech signals and visual face configurations are produced by the human speech organs, these two types of information are strongly correlated and sometimes complement each other. This paper describes two applications based on the relationship between the two types of information, that is, bimodal speech recognition robust to acoustic noise that integrates audio-visual information, and speaking face synthesis based on the correlation between audio and visual speech.

1 Introduction

Speech recognition has seen a drastic progress recently. However, the performance is known to seriously degrade if the system is exposed to a noisy environment. Humans pay attention not only to the speaker's speech but also to the speaker's mouth in such an adverse environment. Lip reading is the extreme case if it is impossible to get any audio signal. This suggests the idea that speech recognition can be improved by incorporating mouth images.

Since both audio speech signals and visual face configurations of the human mouth are produced by the speech organs, these two types of information are strongly correlated and sometimes complement each other. For instance, the recognition accuracy for voiced consonants /b/, /d/, /g/ can be improved by incorporating lip image information, since the lip closure for bilabial phonemes is relatively easy to discriminate using image information. From this point of view, there have been many studies aiming at improving of the speech recognition performance by using lip images [1, 2, 3, 4, 5, 6, 7, 8].

One problem of audio-visual bimodal speech recognition is how to integrate multiple types of information for the recognition of speech, or when the multiple information should be integrated considering the reliability of each piece of information.

The methods studied so far have been classified into two approaches. One is early integration (Direct Integration) and the other is rate integration (Result Integration). The former approach assumes strict dependence and correlation

between audio and visual information. This approach requires a large number of parameters and audio-visual synchronized data to train statistical models. In this approach, acoustic noise interference can seriously degrade recognition performance. On the other hand, the latter approach assumes independence between multiple kinds of information and integrates the result from each recognition system. In this approach the acoustic noise interference does not affect the recognition performance based on visual information, but it can not make use of correlation and is far from human information processing.

This paper presents our methods based on HMMs to incorporate multi-modal information considering synchronization and weights for different modalities. The weights are controlled by the stream weights of the audio-visual HMMs based on the GPD algorithm, in order to adaptively optimize the audio-visual ASR. In particular, we examine the amount of data necessary to estimate the optimum stream weights. This paper also describes one more application regarding audio-visual integration. If face speaking movements can be synthesized well enough for natural communications, a lot of benefits can be brought into the human-machine communications. In addition, we introduce HMM-based speech driven lip movement synthesis based on correlation between audio-visual speech information.

This paper describes audio-visual bimodal speech recognition in section 2 and speech-to-lip movement synthesis in section 3.

2 Audio-Visual Speech Recognition

Audio-visual speech recognition has the possibility of improving the conventional speech recognition performance incorporating visual lip information. The speech recognition performance is degraded in an acoustically noisy environment, unlike visual information. However, visual information itself is insufficient for building a speech recognition system since its phonetic discriminative performance is poor.

As an audio-visual integration method for speech recognition, two approaches have been studied. One is early integration (Direct Integration) and the other is rate integration (Result Integration). In the early integration scheme for HMM-based audio-visual speech recognition, only one set of HMMs is used and the output probability is obtained by multiplying the output probabilities of the audio and visual streams as follows,

$$b(o_t) = b_A(o_t^A)^{\alpha_A} \cdot b_V(o_t^V)^{\alpha_V}, \quad (1)$$

$$(\alpha_A + \alpha_V = 1.0, A : \text{Audio}, V : \text{Visual})$$

where $b(o_t)$, $b_A(o_t^A)$, and $b_V(o_t^V)$ are the output probabilities at time t for the audio-visual, audio, and visual streams, respectively. α_A and α_V are stream weights for audio and visual streams. The recognition performance can be improved further by optimizing stream weights for the output probabilities of the audio and visual streams. This approach is based on dependence and correlation

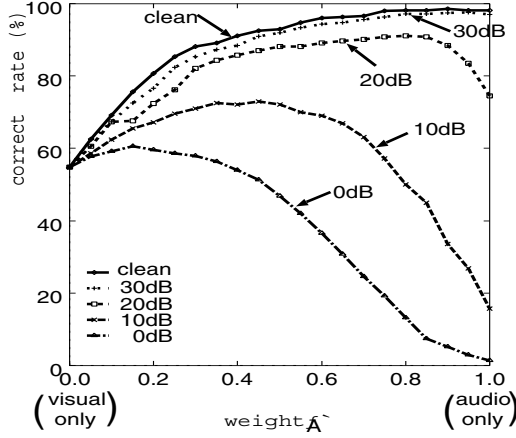


Fig. 1. Effects of stream weights in bimodal speech recognition.

between audio and visual information. A large number of parameters and audio-visual synchronized data for model training are required to represent audio-visual coincident events.

On the other hand, in the rate integration scheme for the HMM-based speech recognition, two kinds of HMM a posteriori probabilities for each input utterance are obtained by,

$$P(O|M) = P_A(O^A|M_A)^{\alpha_A} \cdot P_V(O^V|M_V)^{\alpha_V}, \quad (2)$$

$$(\alpha_A + \alpha_V = 1.0, A : \text{Audio}, V : \text{Visual})$$

where $P(O|M)$, $P_A(O^A|M_A)$, and $P_V(O^V|M_V)$ are the a posteriori probabilities for the observation sequences for the audio-visual, audio, and visual streams, respectively. α_A and α_V are the stream weights of the audio and visual streams. Figure 1 shows experiment results of speaker-dependent 100-word audio-visual isolated word recognition based on early integration. The curves are obtained by changing the stream weight of the audio information. Recognition rate peaks can be observed between the audio-only and visual-only conditions in almost all of the acoustic SNR environments. These peaks indicate the effects of integrating different types of information.

2.1 Audio-Visual Integration Based on Product HMM

In the early integration scheme, a conventional HMM is trained using a fixed amount of audio-visual data. This method, however, cannot sufficiently represent coincident events between the audio and visual information. Furthermore, the visual features of the conventional audio-visual HMM may end up relatively poorly trained because of mis-alignments during model estimation caused by

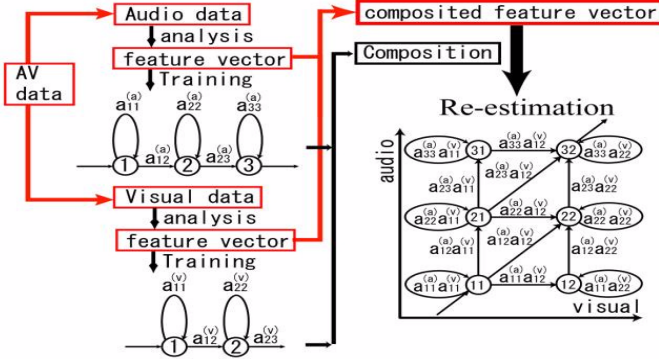


Fig. 2. Training for a product HMM.

segmentation of the visual features. In the late integration scheme, audio data and visual data are processed separately to build two independent HMMs. This scheme assumes asynchronization between the audio and visual features. In addition, it can make the best use of the audio and visual data because of a smaller bi-modal database than the typical database for audio only.

In this paper, in order to model the synchronization between audio and visual features, we propose a method of audio-visual integration based on HMM composition, and formulate a product HMM as a Multi-Stream HMM.

Figure 2 shows the proposed training algorithm. First, in order to create audio-visual phoneme HMMs, audio and visual features are extracted from audio-visual data. In general, the frame rate of audio features is higher than that of visual features. Accordingly, the extracted visual features are incorporated such that the audio and visual features have the same frame rate. Second, the audio and visual features are modeled individually into two HMMs by the EM algorithm. The audio-visual phoneme HMM is composed as the product of these two HMMs. The output probability at state ij of the audio-visual HMM is,

$$b_{ij}(O_t) = b_i^A(O_t^A)^{\alpha_A} \times b_j^V(O_t^V)^{\alpha_V} \quad (3)$$

which is defined as the product of the output probabilities of the audio and visual streams. Here, $b_i^A(O_t^A)^{\alpha_A}$ is the output probability of the audio feature vector at time instance t in state i , $b_j^V(O_t^V)^{\alpha_V}$ is the output probability of the visual feature vector at time instance t in state j , and α_A and α_V are the audio stream weight and visual stream weight, respectively. In general, since equation (3) does not represent a probability mass function, it is improper to estimate the stream weights by the ML principle [9, 10]. In a similar manner, the transition probability

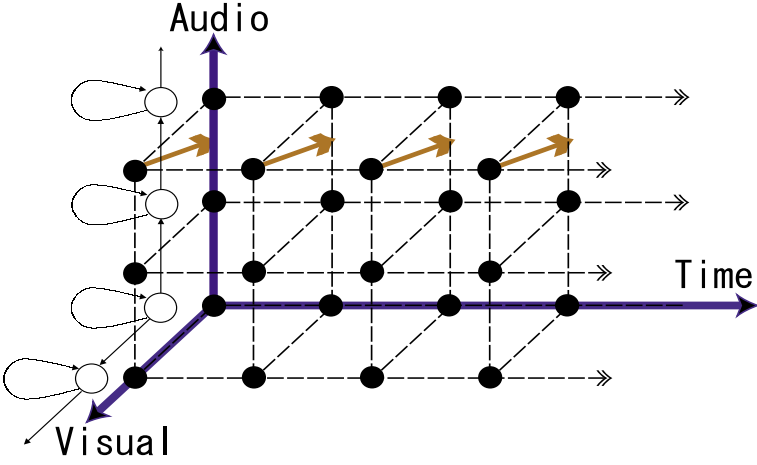


Fig. 3. Trellis Search Space.

from state ij to state kl in the audio-visual HMM is defined as follows,

$$a_{ij,kl} = a_{ik}^A \times a_{jl}^V \quad (4)$$

where a_{ik}^A is the transition probability from state i to state k in the audio HMM, and a_{jl}^V is the transition probability from state j to state l in the visual HMM. This composition is performed for all phonemes.

In the method proposed by [17], a similar composition is used for the audio and visual HMMs. However, because the output probabilities of the audio and visual features are not formulated as a Multi-Stream HMM [8], it is difficult to adaptively optimize the composed HMM for the environment. Furthermore, because the audio and visual HMMs are trained individually, the dependencies between the audio and visual features are ignored. In consideration of this, we re-estimate the composed HMM with the audio-visual feature vectors by the EM algorithm. Then, the probability for the synchronization of the audio and visual features is trained by the above re-estimation. In this paper, in order to verify the effect of the re-estimation, we conduct experiments for two cases, i.e., whether the composed HMM is re-estimated or not.

When speech is recognized by the composed HMM, the search space becomes a three-dimensional HMM trellis with an audio axis, visual axis and time axis (See Fig. 8). This allows the asynchronous temporal movements observed between audio and visual features within a phoneme, and can represent the asynchronization between audio and visual features.

Figure 4 illustrates forced state alignment results. The four lines show the state alignments for audio data using audio HMMs, visual data using visual HMMs, audio-visual data using product HMMs without re-estimation, and audio-visual data using product HMMs with re-estimation. Different state alignments

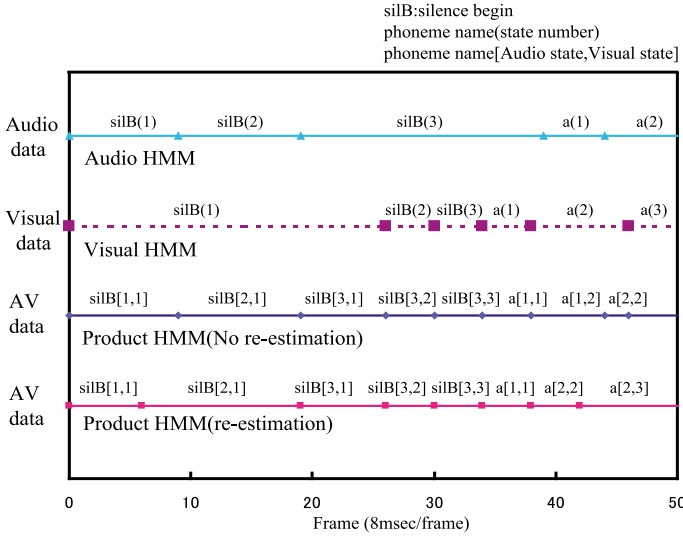


Fig. 4. State Alignments for /au/.

are observed for the audio only and visual only data. In order to represent this combination by early integration, a large number of states is required. The product HMMs try to align states to the audio-visual data based on the product HMMs. However, the model induces a problem that each state of audio and visual HMMs is assigned to the same time period. By re-estimating the product HMMs, this inconsistency can be solved. The model parameters of the product HMMs can be estimated by considering time consistent segmentation.

2.2 Adaptive Estimation of the Stream Weights

As methods for estimating stream weights, maximum likelihood [15] based methods or GPD (Generalized Probabilistic Descent) [16] based methods have been proposed. However, the former methods have a serious estimation drawback because the scales of two probability are normally very different and so the weights can not be estimated optimally. The latter methods have substantial possibility for optimizing the weights. However, a serious problem is that these methods require a lot of adaptation data is necessary for the weight estimation. In this paper, we examine adaptive estimation of stream weights based on the GPD algorithm for new noisy acoustic conditions.

The approach by the GPD training defines a misclassification measure, which provides distance information concerning the correct class and all other competing classes. The misclassification measure is formulated as a smoothed loss function. This loss function is minimized by the GPD algorithm. Here, let $L_c^{(x)}(A)$

be the log-likelihood score in recognizing input data x for adaptation using the correct word HMM, where $\Lambda = \{\lambda_A, \lambda_V\}$.

In a similar way, let $L_n^{(x)}(\Lambda)$ be the score in recognizing data x using the n -th best candidate among the mistaken word HMMs.

The misclassification measure is defined as,

$$d^{(x)} = -L_c^{(x)}(\Lambda) + \log\left[\frac{1}{N} \sum_{n=1}^N \exp\{\eta L_n^{(x)}(\Lambda)\}\right]^{\frac{1}{\eta}} \quad (5)$$

where η is a positive number, and N is the total number of candidates. The smoothed loss function for each data is defined as,

$$l^{(x)} = [1 + \exp\{-\alpha d^{(x)}(\Lambda)\}]^{-1} \quad (6)$$

where α is a positive number. In order to stabilize the gradient, the loss function for the entire data is defined as,

$$l(\Lambda) = \sum_{x=1}^X l^{(x)}(\Lambda) \quad (7)$$

where X is the total amount of data. The minimization of the loss function expressed by equation (7) is directly linked to the minimization of the error. The GPD algorithm adjusts the stream weights recursively according to,

$$\Lambda_{k+1} = \Lambda_k - \varepsilon_k E_k \nabla l(\Lambda), k = 1, \dots, \quad (8)$$

where $\varepsilon_k > 0$, $\sum_{k=1}^{\infty} \varepsilon_k = \infty$, $\sum_{k=1}^{\infty} \varepsilon_k^2 < \infty$, and E is a unit matrix. The algorithm converges to a local minimum as $k \rightarrow \infty$ [13].

2.3 Noisy Speech Recognition Experiments

We conducted experiments to evaluate product HMMs with the adaptive weight optimization algorithm. The experiments are explained in the following. The audio signal is sampled at 12 kHz (down-sampled) and analyzed with a frame length of 32 msec every 8 msec. The audio features are 16-dimensional MFCC and 16-dimensional delta MFCC. On the other hand, the visual image signal is sampled at 30 Hz with 256 gray scale levels from RGB. Then, the image level and location are normalized by a histogram and template matching. Next, the normalized images are analyzed by two-dimensional FFT to extract 6x6 log power 2-D spectra for audio-visual ASR [7]. Finally, 35-dimensional 2D log power spectra and their delta features are extracted.

For each modality, the basic coefficients and the delta coefficients are collectively merged into one stream. Since the frame rate of the video images is 1/30, we insert the same images so as to synchronize the face image frame rate to the audio speech frame rate.

For the HMMs, we use two mixture Gaussian distribution and assign three states for the audio stream and two states for the visual stream in the late

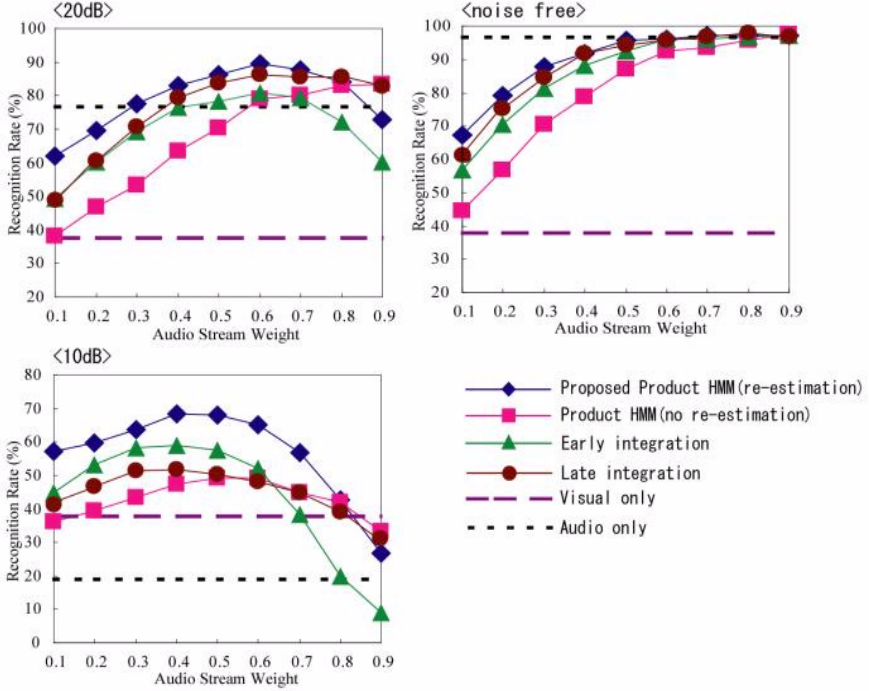


Fig. 5. Recognition results for various stream weights.

integration HMMs and product HMMs. In contrast, three states are assigned for early integration. In this research, we perform word recognition evaluations using a bi-modal database [8]. We use 4740 words for HMM training and two sets of 200 words for testing. These 200 words are different from the words used in the training.

In an experiment to adaptively optimize the stream weights, we use 100 words as the adaptation data, excluding the training and test data. We also perform experiments using 15, 25, and 50 words, which are extracted from the 100 words. The context of the data for the adaptation differs from that of the test data. In order to examine in more detail the estimation accuracy in the case of less adaptation data, we carry out recognition experiments using three sets of data, each as different as possible from the context. The size of the vocabulary in the dictionary is 500 words during the recognition of the adaptation data. The GPD algorithm convergence pattern is known to greatly depend on the choice of parameters. Accordingly, we set $N = 1$ in (3), $N = 0.1$ in (4), $N = 100/k$, and the maximum the iteration count = 8.

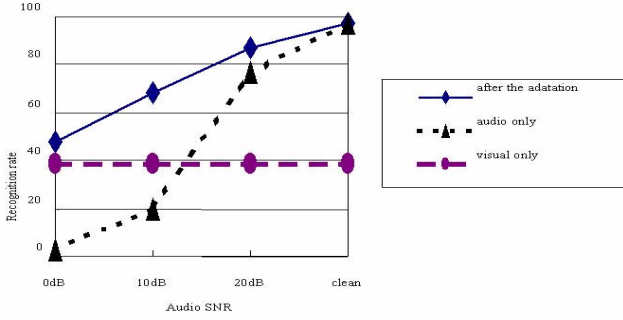


Fig. 6. Recognition results with optimized weights.

Figure 5 shows recognition rates for stream weights for audio only, visual only, early integration, late integration and the proposed integration (re-estimated/no re-estimation). In the figure, the stream weights change under $\lambda_A + \lambda_V = 1$ in the acoustically noisy audio SNR 10dB, 20dB and noise free environments.

Figure 5 proves that the recognition rates of the proposed method are higher than those of the other methods. The reasons for this are as follows. The early integration cannot represent the asynchronization between audio and visual features very well, and the late integration regards each feature as independent. In the proposed method, on the other hand, the probability for the synchronization of the audio and visual features is trained. Product HMMs can also represent the asynchronization between the data by searching a three-dimensional HMM trellis. When the number of initial HMM states is simply increased, the parameters are not estimated well and the recognition rate worsens. For these reasons, we can consider that a better initial model can be created by composing audio and visual HMMs. In addition, even if product HMMs cannot be re-estimated because of a lack of data, they can be used in the decoding as is.

Generally, audio-visual ASR systems have their peak recognition rates by stream weights (See Fig. 4). Therefore, if the optimal stream weights can be estimated by a small amount of adaptation data, audio-visual ASR systems can achieve higher recognition rates in various environments. Figure 5 also shows an average of estimated stream weights from 25 word data by the GPD algorithm.

The average of the estimated stream weights is almost optimized in Figure 5. However, when the amount of adaptation data is small, the estimated stream weights are sometimes different from the optimal values because of the difference in the data context. Therefore, we examine the standard deviation of the estimated stream weights and show the recognition rate by three sets of data in Table II. Each standard deviation in the table is the average for noise free, 20dB, 10dB, or 0dB. Table II shows that the larger the amount of adaptation

data is, the lower the standard deviation is. If there are more than 25 words for the adaptation, the a near optimal value is estimated.

Figure 1 shows recognition rates for audio SNR. Note that the performance of the proposed ASR system is better than that of the uni-modal one, even in the noise free case.

Table 1. Standard deviation for the amount of data.

The amount of adaptation data	Standard deviation of stream weight	Standard deviation of recognition rate
15 words	0.1330	7.7397
25 words	0.0970	1.9566
50 words	0.0804	0.7514

3 Speech-to-Lip Movement Synthesis Based on HMMs

Lip-synchronization is required for human-like computer agents in interactive communication systems. If lip movements can be synthesized well enough to do lip-reading, hearing impaired people can compensate for their inability to obtain auditory information through the computer agents. In this section, speaking face synthesis is described as an example to make use of correlation between audio and visual speech signals.

Mapping algorithms from speech to lip movements have been reported based on: Vector Quantization [17], Artificial Neural Networks [18, 19, 22], and Gaussian Mixtures [22]. These methods are based on frame-by-frame or frames-by-frames mapping from speech parameters to lip parameters. However, these mapping algorithms have problems, such that 1) the mapping is fundamentally a complicated many-to-many mapping, and 2) extensive training is required to take account of context information. The required audio-visual database increases in proportion to the length over the preceding or succeeding frames.

On the other hand, there are other approaches that use techniques of speech recognition, such as phonetic segmentation [23] and HMM [21, 24, 22, 25]. These methods convert speech into lip parameters based on information such as a phonetic segment, a word, a phoneme, or an acoustic event. The HMM-based methods have an advantage in that explicit phonetic information is available to help one consider co-articulation effects caused by surrounding phoneme contexts.

This paper describes a new method to estimate visual parameters by applying the Expectation-Maximization algorithm (MAP-EM). The MAP-EM method repeatedly estimates visual parameters while maximizing the likelihood of the audio and visual joint probability of audio-visual HMMs. The re-estimating operation is regarded as the auto-association of a complete pattern out of an incomplete pattern for a time series. In experiments, the MAP-EM method is compared to the MAP-V method.

3.1 Mapping Method Using Viterbi Algorithm

The first method is our baseline method, MAP-V^[25], which is composed of two processes. The first process is a decoding process that converts the given speech to the most likely HMM state sequence by the algorithm and the second is a look-up table process that converts an HMM state to corresponding visual parameters.

In the decoding process, the likelihood of the input speech by the k -th audio HMM, M_k^A , is defined as,

$$P(O^A|M_k^A) \approx \max_Q \{ \pi_{q_1}(M_k^A) \times \prod_{t=1}^T a_{q_{t-1}q_t}(M_k^A) b_{q_t}(o^A(t)|M_k^A) \}, \quad (9)$$

where $Q = q_1 \cdots q_T$ denotes the audio HMM state sequence, $O^A = o^A(1) \cdots o^A(T)$ is the sequence of input audio parameters, $\pi_j(M_k^A)$ is the initial state probability, $a_{ij}(M_k^A)$ is the transition probability from state i to j of M_k^A , and $b_j(o^A(t)|M_k^A)$ is the output probability at state j .

In equation (9), the optimal state sequence for the observation is obtained by Viterbi alignment. Along the alignment, the correspondence between each audio HMM state and the visual parameters is then calculated and stored in the look-up table in the training step. The visual parameters per audio HMM state are obtained by taking the average of all visual parameters assigned to the same audio HMM state.

3.2 New Mapping Method Using the EM Algorithm

Estimating Visual Parameters by Auto-association. The quality of visual parameters by the MAP-V method depends on the accuracy of the Viterbi alignment. Incorrect HMM states by the Viterbi alignment may produce wrong lip images. The proposed MAP-EM method does not depend on the deterministic Viterbi alignment.

The proposed method re-estimates the visual parameters for the given audio parameters by the EM algorithm using audio-visual HMMs. Although the visual parameters do not exist initially, the required visual parameters are synthesized iteratively from the initial parameters by the re-estimation procedure maximizing the likelihood of audio-visual joint probability of the audio-visual HMMs.

$$\hat{o}^V(t) = \arg \max_{o^V(t)} P(O^{A,V}|o^A(t), M_k^{AV}), \quad (10)$$

where $\hat{o}^V(t)$ means the estimated visual parameters. The likelihood of the proposed method is derived by considering all HMM states at a time. To treat all

states of all HMMs evenly, the likelihood of the audio-visual joint probability is defined as follows,

$$\begin{aligned}
& \sum_{Q(all\ k)} P(M_k^{AV})P(O^{A,V}|Q, M_k^{AV}) \\
&= \sum_{Q(all\ k)} P(M_k^{AV})\pi_{q_1}(M_k^{AV}) \\
&\quad \times \prod_{t=1}^T a_{q_{t-1}q_t}(M_k^{AV})b_{q_t}(o^{A,V}(t)|M_k^{AV}), \tag{11}
\end{aligned}$$

where $P(M_k^{AV})$ is the probability of the k -th HMM, and $\pi_j(M_k^{AV})$, $a_{ij}(M_k^{AV})$ and $b_j(o^{A,V}(t)|M_k^{AV})$ are the joint initial state probability, joint transition probability, and joint output probability of the audio and visual parameters, respectively. The summation of $Q(all\ k)$ considers all models M_k^{AV} at a time. The next section describes the derivation of the re-estimation formula of visual parameter.

Algorithm of Visual Parameter Estimation. The re-estimation formula is defined to maximize auxiliary function $A(\hat{o}^V(t), o^V(t))$ over estimated visual parameter $\hat{o}^V(t)$.

$$\begin{aligned}
& A(\hat{o}^V(t), o^V(t)) \\
&= \sum_{Q(all\ k)} P(M_k^{AV})P(O^{A,V}|Q, M_k^{AV}) \\
&\quad \times \log P(M_k^{AV})P(\hat{O}^{A,V}|Q, M_k^{AV}) \tag{12}
\end{aligned}$$

The maximization of the auxiliary function is equivalent to increasing likelihood of the input data. The re-estimation formula of the visual parameters is derived by differentiating the auxiliary function by the m -th visual parameter $\hat{o}_m^V(t)$. Let the output probability density function be mixed Gaussian distributions with mean vectors $\mu_n^A(M_k^{AV}, j)$, $\mu_m^V(M_k^{AV}, j)$ and covariance matrix Σ with its components $\sigma_{nn'}^{A,A}(M_k^{AV}, j)$, $\sigma_{mm'}^{V,V}(M_k^{AV}, j)$, $\sigma_{nm}^{A,V}(M_k^{AV}, j)$. n, m are the numbers of dimensions of the audio and visual parameters. The re-estimation formula is derived as follows,

$$\begin{aligned}
& \hat{o}_m^V(t) \\
&= \frac{1}{\sum_k \sum_j P(M_k^{AV})\gamma(t; M_k^{AV}, j) \frac{\Sigma_{mm'}^{V,V}(M_k^{AV}, j)}{|\Sigma(M_k^{AV}, j)|}} \\
&\quad \times \sum_k \sum_j P(M_k^{AV})\gamma(t; M_k^{AV}, j) \\
&\quad \times \frac{1}{|\Sigma(M_k^{AV}, j)|} (\mu_m^V(M_k^{AV}, j) \Sigma_{mm'}^{V,V}(M_k^{AV}, j) \\
&\quad - \sum_n (o_n^A(t) - \mu_n^A(M_k^{AV}, j)) \Sigma_{nm}^{A,V}(M_k^{AV}, j)), \tag{13}
\end{aligned}$$

where $\gamma(t; M_k^{AV}, j)$ is the state occupation probability in state j of M_k^{AV} at time t . $\Sigma_{mm'}^{V,V}(M_k^{AV}, j)$ means the adjoint of $\Sigma_{mm'}^{V,V}(M_k^{AV}, j)$. Formula (13) is under the constraint that covariance $\sigma_{nn'}^{A,A}(M_k^{AV}, j) = 0$ at $n \neq n'$ and $\sigma_{mm'}^{V,V}(M_k^{AV}, j) = 0$ at $m \neq m'$. Furthermore, the re-estimation formula is simplified as follows if the covariance matrix is diagonal. In this paper, formula (14) is used in experiments.

$$\begin{aligned} \hat{o}_m^V(t) &= \frac{\sum_k \sum_j P(M_k^{AV}) \gamma(t; M_k^{AV}, j) \frac{\mu_m^V(M_k^{AV}, j)}{\sigma_{mm}^{V,V}(M_k^{AV}, j)}}{\sum_k \sum_j P(M_k^{AV}) \gamma(t; M_k^{AV}, j) \frac{1}{\sigma_{mm}^{V,V}(M_k^{AV}, j)}}. \end{aligned} \quad (14)$$

The algorithm for the visual parameter re-estimation can be summarized as the follows:

- step 1.** Set the initial value for visual parameter $o_m^V(t)$.
- step 2.** Calculate $\gamma(t; M_k^{AV}, j)$ for all frames under the Forward-Backward algorithm (EM algorithm for HMMs).
- step 3.** Re-estimate $\hat{o}_m^V(t)$ at each frame.
- step 4.** If the convergence condition is satisfied, go to the end, otherwise, return to step 2.

3.3 Lip Synthesis Experiments

Experiment Conditions. Speech and image data for experiments are synchronically recorded in 125Hz. The visual parameters are height (X), width (Y) of the outer lip contour and protrusion (Z) of the lip sides from an original point, where the three parameters are used to construct a lip shape from 3D-lips software [26]. The audio parameter has 33 dimensions of 16-order mel-cepstral coefficients, their delta coefficients and the delta log power.

Fifty four monophones and two pause models are used for the HMMs in both the MAP-V method and the MAP-EM method. The pause models are prepared separately for the word beginning and the word ending. Triphone HMMs are not adopted, because they require huge amounts of time synchronous training data. Each audio HMM and audio-visual HMM has a left-to-right structure with three states, where the output probability of each state has 256 tied-mixture Gaussian distributions. The HMMs are trained by an audio or audio-visual synchronous database composed of 326 Japanese words, all phonetically balanced. The other one hundred words are prepared for testing.

The measure to evaluate synthesized lip movements is Euclidian error distance E between the synthesized visual parameters and the original parameters extracted from human movements.

In the MAP-EM method, state occupation probabilities $\gamma(t; M_k^{AV}, j)$ are updated after the re-estimation of all visual parameters for an utterance.

Table 2. Compared Synthesis Methods.

method	#params HMM training		#params Mapping		initial visual parameters
	A	V	A	V	
MAP-V	33	—	33	—	—
MAP-EM1	33	3	33	3	MAP-V
MAP-EM2	33	6	33	3	MAP-V
MAP-EM3	33	9	33	3	MAP-V
MAP-EM4	33	9	33	3	pause lip

Table 3. Error distances of synthesis methods.

	E cm			
	all frames	correct decoded	incorrect decoded	incorrect decoded /p//b//m/
MAP-V	1.066	1.062	1.075	1.701
MAP-EM-1	1.093	1.106	1.051	1.370
MAP-EM-2	1.063	1.077	1.021	1.392
MAP-EM-3	1.052	1.072	0.989	1.254
MAP-EM-4	1.207	1.231	1.134	1.061

Compared Synthesis Methods. To verify the effect of the MAP-EM method, the five synthesis methods in table 2 are compared in an experiment. The MAP-EM method can be implemented under various conditions. We try to make the number of parameter vectors fluctuate taking account of the dependency on the quality of HMMs. In the MAP-EM-2 method, the visual parameter vector consists of six parameters of three lip parameters and their time differential parameters. Likewise, the MAP-EM-3 method contains the acceleration part of lip parameters in addition to the parameters of the MAP-EM-2 method. Note that in all MAP-EM methods, the number of tied-mixture distributions is fixed at 256 as well as in the MAP-V method. As for the initial values for the visual parameters, the MAP-EM-1,2,3 methods use the visual parameters synthesized by the MAP-V method and the MAP-EM-4 method uses the visual parameters of the lip closure shape during a pause. The objective evaluation results of the five methods are shown in Table 3. Each column in Table 3 indicates error distances averaged by all frames or correctly decoded, incorrectly decoded, and incorrectly decoded /p//b//m/ frames at the MAP-V method. In the errors averaged by all frames, the MAP-EM-3 method reduces the error distance by 1% against the MAP-V method. The MAP-EM-4 method gives a large error due to the flat start of the lip closure.

We investigate errors of the MAP-EM-1,2,3 methods under incorrectly decoded frames at the MAP-V method. The errors are compared with the three detailed categories of palatal, dental and bilabial consonants in Figure 7. The MAP-V method shows a large error in the bilabial consonant category of incorrectly decoded frames. It is known that the bilabial consonants /p//b//m/ are quite sensitive to audiences. For these phonemes, the errors of the MAP-EM-3

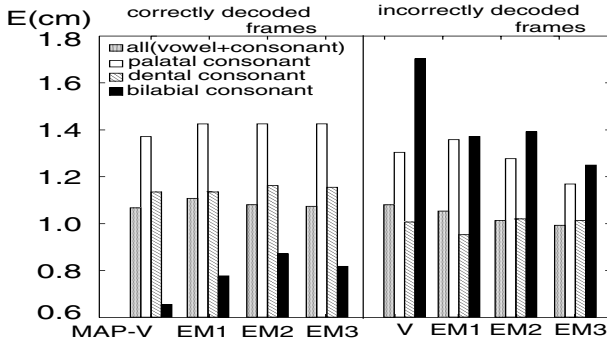


Fig. 7. Errors by consonant category.

method are reduced by 26% compared to the errors of the MAP-V method at incorrectly decoded frames.

An effect of the MAP-EM method is illustrated in Figure 8 and Figure 9. The figures show results for a test Japanese word 'kuchibiru'. The horizontal axis means the number of frames corresponding to time. The vertical axis means visual parameters. The solid lines in the figures are synthesized visual parameters, and the dotted lines are visual parameters by the original recorded human movements. The two vertical lines show the beginning and ending times of the utterance. The synthesized height visual parameter of the MAP-V method does not form a valley for the lip closure of /b/ because of the incorrect Viterbi alignment in Figure 8. However the MAP-EM-3 method of Figure 9 shows the correct articulation.

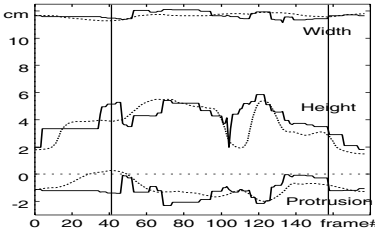


Fig.8 Visual parameters synthesized by MAP-V method.

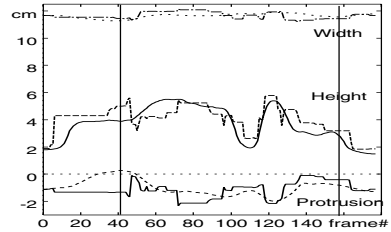


Fig.9 Visual parameters synthesized by MAP-EM-3 method.

4 Conclusion

This paper described two kinds of research we are carrying out for audio-visual bimodal speech processing. Since both audio speech signals and visual face configurations are produced by speech organs, these two types of information are strongly correlated and sometimes complement each other. The first one consists

of the product HMMs for robust speech recognition with stream weight optimization. These product HMMs achieve a effective interaction of audio-visual coincident events. GPD-based stream weight estimation adapts the system rapidly to new environments. The second one consists of image information generation from audio speech signals using HMMs trained with audio-visual synchronized data. The method is fundamentally based on the correlation between audio and visual speech information. The integration of multiple types of information should be one approach towards making the system robust to various environments. Further research is necessary to build efficient and effective fusion methods for integrated multi-modal speech processing.

References

1. Stork, D.G. and Hennecke, M.E.: Speechreading by Humans and Machines. NATO ASI Series, (1995) Springer.
2. Petajan, E.: Automatic Lipreading to Enhance Speech Recognition. Proc. CVPR'85, (1985).
3. Yuhas, B., Goldstein, M., and Sejnowski, T.: Integration of Acoustic and Visual Speech Signals Using Neural Networks. IEEE Communications Mag. (1989) 65–71.
4. Bregler, C., Hild, H., Manke, S., and Waibel, A.: Improving Connected Letter Recognition by Lipreading. Proc. ICSLP'93, (1993).
5. Adjoudani, A. and Benoit, C.: Audio-Visual Speech Recognition Compared Across Two Architectures. Proc. Eurospeech'95, (1995).
6. Silsbee, P.: Computer Lipreading for Improved Accuracy in Automatic Speech Recognition. IEEE Trans. Speech and Audio, (1996) Vol. 4. No. 5.
7. Nakamura, S., Nagai, R., and Shikano, K.: Improved Bimodal Speech Recognition Using Tied-Mixture HMMs and 5000 word Audio-Visual Synchronous Database. Proc. Eurospeech'97, (1997) 1623-1626.
8. Nakamura, S., Ito, H., and Shikano, K.: Stream Weight Optimization of Speech and Lip Image Sequence for Audio-Visual Speech Recognition. Proc. ICSLP'2000, (2000), Vol. 3, 20–23.
9. Potamianos, G. and Graf, H.P.: Discriminative Training of HMM Stream Exponents for Audio-Visual Speech Recognition. Proc. ICASSP'98, (1998), 3733-3736.
10. Miyajima, C., Tokuda, K., and Kitamura, T.: Audio-Visual Speech Recognition Using MCE-based HMMs and Model-dependent Stream Weights. Proc. ICSLP'2000, (2000), 1023-1026.
11. Duchnowski, P., Meier, U., and Waibel, A.: See Mee, Hear Me: Integrating Automatic Speech Recognition and Lip-Reading. Proc. ICSLP'94, (1994).
12. Tomlinson, M., Russell, M., and Brooke, N.: Integrating Audio and Visual Information to Provide Highly Robust Speech Recognition. Proc. ICASSP'96, (1996).
13. Katagiri, S., Juang, B-H., and Lee, C-H.: Pattern Recognition using a Family of Design Algorithms based upon the Generalized Probabilistic Descent Method. Proc. IEEE, (1998) Vol. 86, No. 11.
14. Alissali, M., Deleglise, P., and Rogozan, A.: Asynchronous Integration of Visual Information in an Automatic Speech Recognition System. Proc. ICSLP'96, (1996).
15. Hernando, J.: Maximum Likelihood Weighting of Dynamic Speech Features for CDHMM Speech Recognition. Proc. ICASSP'97C(1997) 1267–1270.
16. Potamianos, G. and Graf, H.P.: Discriminative Training of HMM Stream Exponents for Audio-visual Speech Recognition. Proc. ICASSP'98C(1998) 3733–3736.

17. Morishima, S., Aizawa, K., and Harashima, H.: An Intelligent Facial Image Coding Driven by Speech and Phoneme. Proc. ICASSP'89, (1989), 1795–1798.
18. Morishima, S. and Harashima, H.: A Media Conversion from Speech to Facial Image for Intelligent Man-Machine Interface. IEEE Journal on sel. areas in Communications, (1991) Vol. 9, No. 4, 594–600.
19. Lavagetto, F.: Converting Speech into Lip Movements: A Multimedia Telephone for Hard of Hearing People. IEEE Trans. on Rehabilitation Engineering, (1995) Vol. 3, No. 1, 90–102.
20. Simons, A. and Cox, S.: Generation of Mouthshape for a Synthetic Talking Head. Proc. The Institute of Acoustics, (1990) Vol. 12, No. 10.
21. Chou, W. and Chen, H.: Speech Recognition for Image Animation and Coding. Proc. ICASSP'95 (1995) 2253–2256.
22. Rao, R.R. and Chen, T.: Cross-Modal Prediction in Audio-Visual Communication. Proc. ICASSP'96, (1996) Vol. 4, 2056–2059.
23. Goldenthal, W., Waters, K., Van Thong, J.M., and Glickman, O.: Driving Synthetic Mouth Gestures: Phonetic Recognition for FaceMe!. Proc. Eurospeech'97 Vol. 4, (1997) 1995–1998.
24. Chen, T. and Rao, R.: Audio-Visual Interaction in Multimedia Communication. Proc. ICASSP'97, (1997) 179–182.
25. Yamamoto, E., Nakamura, S., and Shikano, K.: Speech-to-Lip Movement Synthesis by HMM. ESCA Workshop of Audio Visual Speech Processing, (1997) 137–140.
26. Guiard-Marigny, T., Adjoudani, T., and Benoit, C.: A 3-D model of the lips for visual speech synthesis. Proc. of the Second ESCA/IEEE Workshop on Speech Synthesis (1994).

Revisiting Carl Bildt’s Impostor: Would a Speaker Verification System Foil Him?

Kirk P.H. Sullivan¹ and Jason Pelecanos²

¹ Umeå University, 901 87 Umeå, Sweden
`kirk@ling.umu.se`

² Speech Research Lab, Queensland University of Technology
GPO Box 2434, Brisbane, Australia, 4001
`j.pelecanos@qut.edu.au`

Abstract. Impostors pose a potential threat to security systems that rely on human identification and verification based on voice alone and to security systems that make use of computer audio-based person authentication systems. This paper presents a case-study, which explores these issues using recordings of a high quality professional impersonation of a well-known Swedish politician. These recordings were used in the role of impostor in the experiments reported here. The experiments using human listeners showed that an impostor who can closely imitate the speech of the target voice can result in confusion and, therefore, can pose a threat to security systems that rely on human identification and verification. In contrast, an established Gaussian mixture model based speaker identification system was employed to distinguish the recordings. It was shown that the recognition engine was capable of classifying the mimic attacks more appropriately.

1 Introduction

The robustness of speaker verification systems are traditionally assessed by experimentally using impostors selected at random from an exclusive set of speakers in a database. A notable exception to this is the study reported in [3]. In this study the impostor was not selected from the other speakers in the database, but rather, use was made of concatenation of client speech, formant copy synthesis of client speech utterances and client-based diphone synthesis to ‘create’ the impostor. The underlying premise of the study was that the impostor would know who they were attempting to mimic. This diverges from the majority of earlier speaker verification studies in which it was assumed that an impostor would not be attempting to be any particular person in the database (but rather trying their luck to break into the security system). The authors wished to test a speaker verification system using the worse case scenario in which the impostor has access to some client speech samples. Not surprisingly, the study found the concatenation of client speech to be a very effective means of creating a coherent recording that could be passed off as the client’s speech. The remaining two approaches were not successful.

The case study presented here follows the premise of [3], but rather than using a range of speech synthesis techniques to attack a speaker verification system, this paper makes use of high-quality professional audio imitations of a single client voice. By assessing how well this particular professional imitator is at attacking a speaker verification system, this case study revisits a similar question addressed in [6]: Are untrained human listeners able to detect the professional imitations of this particular client's voice? The range of possible tasks, in this initial study of the impact of high quality professional imitation is delimited by the set of recordings used in [5]. Due to the demands of the speaker identification system used here, a subset of this data is analyzed. This work extends the research conducted in [3], by investigating whether speaker recognition systems are vulnerable to voice imitation, but also juxtaposes their detection characteristics with the ability of human listeners to detect high quality imitations of this particular client's voice. The recordings used in [5] and this case study facilitate in part the answering of these questions with respect to a single client voice.

2 The Recordings

The client recordings consisted of two approximately 30 second long excerpts of uninterrupted natural speech by the former *Statsminister* (Prime Minister) of Sweden, Carl Bildt (CB). These are identified as the spoken Passages A and B. Based upon these recordings, a second set of recordings were made by a professional Swedish impersonation artist (IB). This impersonator attempted to imitate the speaker in these excerpts as closely as possible. These imitations were not intended to be entertaining, but rather explicitly meant to mimic the client voice and speech style. A second recording of each of the client recordings was made by the imitator in his natural voice and speaking style (NI). Further recordings were made by four amateur imitators.

3 Evaluation of the Imitations by Human and Machine

A telephony speech study using the National Institute of Standards and Technology (NIST) 1998 Speaker Recognition Evaluation [6] compared the performance of automated speaker verification systems with that of humans. Interestingly, it was found that the mean human opinion (using typically 16 listeners) performed better than the automated systems for the same and different telephone number conditions. Human listeners performed significantly worse when evaluated individually. It is also noted that in general, the relative drop in performance when comparing the matched to the mismatched training to testing conditions was the same or greater for the algorithmic approaches than for the mean human evaluation. This study investigated the aspect of random impostors and did not evaluate the corresponding performances for mimic attacks. In this paper, we evaluate for a single case study, the characteristics of the human and machine evaluation of mimic attacks as it applies to speaker identification. The experiments were conducted as voice line-ups as specified in [5].

3.1 Line-Up Construction and the Human Evaluation Tasks

The recordings from Passage A, for the purpose of a voice line-up, were divided into fourteen circa two-second sentences. Of these fourteen groups, the sentences, “*och därför tycker jag att det är så underligt*”, “and therefore I find it so strange” (Sentence A) and “*att där vill miljöpartiet bromsa*”, “that, there the Green Party wants to drag its heels” (Sentence B) were selected as the stimuli to be used in the voice line-ups. The basis for the selection of these stimuli is presented in [Fig. 1](#).

The voice line-ups comprised six voices played consecutively with a short pause between each sentence. Voice line-ups were constructed with and without the client’s voice (CB), with and without IB and with and without NI. Additional stimuli to complete the speaker line-ups, and act as foils, were randomly selected from the set of amateur recordings. A total of sixteen different voice line-ups were created; the order of presentation within each voice line-up was random. Only those line-ups that contained the target voice were compared in this human and machine experiment.

The study in [Fig. 1](#) conducted four different tasks; two of these tasks are of interest here. For both tasks, ten female and five male listener participants heard the passage B recording of the voice they were to identify, twice before a training block of line-ups and twice before the experimental block of sixteen voice line-ups. The participants were told that they would hear a target voice, and that they attempt to recognize the speaker (if present) in the voice line-ups that followed. In one task, the target voice was CB. Here we are interested in how often the imitation in the line-up is selected as CB. This would indicate how easily human listeners are convinced that IB is CB and would, therefore, threaten a security system relying on human opinion. In the remaining task, the target voice was IB. Here, we are interested in how well the imitator has disguised his natural voice in the line-up. All the participants indicated after the experiment that they were familiar with the voice of CB and not familiar with the voices of any of the other speakers.

4 Automatic Speaker Identification

In this task we examine the application of a closed-set automatic speaker identification system to the Carl Bildt impostor attack problem. Automatic speaker identification is the process of identifying a possible client or target speaker from a list of possible line-up or candidate speakers. In closed-set identification, the target speaker is always present in the speaker line-up.

In this speaker recognition system, there exist three processes; parameterization, speaker modeling and speaker classification. The parameterization phase compresses the speech into a more compact form, termed features, which are then usable by the classifier. The speaker modeling process models the features derived from a particular target speaker of interest. The classification task tests an utterance against a list of speaker models to determine the most likely candidate speaker.

4.1 Parameterization

The speech is parameterized using 20 filterbanks to form 12 Mel-Frequency Cepstral Coefficients (MFCCs) [1] for a frame size of 32ms and a shift of 10ms. The corresponding delta coefficients were included. Cepstral mean subtraction was performed on the 12 base coefficients to limit the linear channel effects. An energy based silence removal scheme was used.

4.2 Gaussian Mixture Modeling

Gaussian Mixture Modeling (GMM) is used for modeling the Probability Density Function (PDF) of a set of multi-dimensional feature vectors. A GMM forms a continuous probability density estimate of the multi-variate parameters by the additive composition of multi-dimensional Gaussians. Given a single speech feature vector \mathbf{x} of dimension D , the probability density of \mathbf{x} given an N Gaussian mixture speaker model λ , with mixture weights w_i , means $\boldsymbol{\mu}_i$, and covariances $\boldsymbol{\Sigma}_i$, is given by $p(\mathbf{x}|\lambda) = \sum_{i=1}^N w_i g(\mathbf{x}, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ with a single Gaussian component density given as

$$g(\mathbf{x}, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) = \frac{1}{(2\pi)^{\frac{D}{2}} |\boldsymbol{\Sigma}_i|^{\frac{1}{2}}} \exp \left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)' (\boldsymbol{\Sigma}_i)^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) \right) \quad (1)$$

where $(\cdot)'$ represents the matrix transpose operator.

To model the distribution of a set of training vectors, an iterative method is used. The GMM parameters are established by use of the Maximum-Likelihood estimate using the Expectation-Maximization algorithm proposed by Dempster [2]. The GMM specific result can be located in [1]. For this experiment, a GMM was formed from 40 Gaussian mixture components.

4.3 Speaker Classification

Speaker scoring is determined by finding the log-likelihood for each test utterance and speaker model comparison. Given a set of T feature vectors from an utterance $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T\}$, the test utterance log-likelihood given the target model (assuming independent and identically distributed observations) can be determined by $\log p(X|\lambda) = \sum_{t=1}^T \log p(\mathbf{x}_t|\lambda)$. Given an utterance X , to be tested against a set of K line-up speaker models $\{\lambda_1, \lambda_2, \dots, \lambda_K\}$, the most likely speaker index S (assuming equal prior speaker probabilities), can be determined by $S = \arg \max_{1 \leq k \leq K} \log p(X|\lambda_k)$. Thus, given a test utterance, the model with the highest likelihood is the hypothesized target speaker.

4.4 Adaptation to Human Listening Experiments

The automatic speaker identification system was to be configured to match the conditions mentioned in the human evaluation. This poses a problem in the context of the human evaluation experiment in that there is insufficient training

data (approximately 2 seconds of speech) to form speaker models from the line-up of speakers. Thus, a different configuration of this system to determine the most likely candidate was proposed. Instead of attempting to create inaccurate speaker models from the 2 seconds of line-up speech, the natural voices of all the candidates were used in establishing speaker models. Then, for a single line-up test, each of the six test segments were compared against the speaker models. To determine which speaker from the line-up is the speaker of interest, the requirement is that only one of the line-up extracts can designate the target model as the most likely candidate from a list of all the natural voice models. Given the experimental configuration, any other possibilities then default to either an indecisive or incorrect system result. Thus, this method of performing speaker identification with limited test data relies upon the effectiveness of the impostor models, corresponding to speakers in the line-up, to eliminate the unlikely candidates from the test recording.

5 Results and Discussion

The closed-set automatic speaker identification system as applied to the Carl Bildt impostor attack problem was evaluated using the same line-ups constructed and presented to the human listeners. (Speaker models were trained using passage B with tests from passage A.) For the task when CB was the client voice, the speaker identification system selected this voice from the line-up as the most likely candidate by a significant margin. Thus, IB failed to be selected, as did NI or any of the foils. This was true for both Sentences A and B. This human result was marginally worse, with four listener responses out of 60 selecting IB when the line-ups contained both CB and IB. When IB was not present in the line-up, CB was selected correctly each time except for two that specified the not present option.

For the identification system, when NI was the client voice, the results were not as pronounced as for when CB was the client voice. For Sentence B, NI was consistently selected, yet for Sentence A it was never selected. A mixed opinion of speaker foils and CB were selected as the more likely voices. However, the NI segment extracted from the line-up did not select any particular target model significantly over another in the automatic system, which emphasizes the uncertainty tied with the NI segment from the line-up. Interestingly, this result differs from the human listener responses where Sentence A was selected 58% of the time and Sentence B 50% of the time. As with the automatic system, no particular voice was consistently selected for the other responses.

A second set of more significant test segments using the entire 30 seconds of Passage A was also conducted on the automatic speaker identification system. Here, because there was sufficient data for the test segments, the standard speaker identification configuration could be used. For these tests there are no human listener equivalent perception tests. The outcome when CB was the test voice, was that CB was selected as the most likely candidate speaker, followed by NI and then the professional imitation, IB. The outcome when NI was the

client voice was that NI was selected as the most likely candidate voice, followed by IB and then finally, CB.

It is proposed that when using speaker models for both CB and NI that Göran Gabrielsson's imitation attack, IB, is more likely to be verified as his own voice, NI, than that of Carl Bildt, CB. In all cases, Göran Gabrielsson's imitation attack, IB, failed to be recognized as the voice of Carl Bildt, CB, but was recognized as Gabrielsson's own natural voice, NI. This result differs greatly from the human listener outcome. For the line-up containing CB and NI, but not IB, when the listeners were asked to identify IB, 90% of the selections were of CB and the remaining 10% claimed that the voice was not present in the line-up. No human listener selected NI as the voice they had been asked to remember and identify.

6 Conclusion

In the context of this case study, there were a number of observable results. The speaker verification system (in this case study) was capable of foiling the Gabrielsson impostor attacks on Carl Bildt by indicating that the impostor, IB, was more likely to be Gabrielsson, NI. The human listeners were poorer performers in this regard. In addition, it was found that the recognition system could not consistently specify the true identity of the imitator after he modified his voice. Thus, the impostor attacks on Bildt were foiled, but there was little evidence to suggest that Gabrielsson was the impostor.

References

1. Davis, S. and Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences In: *IEEE Transactions on Acoustic Speech Signal Processing*, Vol. ASSP-28, (1980) 357–366.
2. Dempster, A., Laird, N., and Rubin, D.: Maximum likelihood from incomplete data via the E-M algorithm In: *Journal of the Royal Statistical Society B*, Vol. 39, No. 1, (1977) 1–38.
3. Lindberg, J. and Blomberg, M.: Vulnerability in speaker verification — a study of possible technical imposter techniques. In: *Proceedings of the 6th European Conference on Speech Communication and Technology*, Vol. 3. (1999) 1211–1214.
4. Reynolds, D. and Rose, R.: Robust Text-Independent Speaker Identification Using Gaussian Mixture Models In: *IEEE Transactions on Speech and Audio Processing*, Vol. 3, No. 1, (1995) 72–83.
5. Schlichting, F. and Sullivan, K.P.H.: The imitated voice — a problem for voice line-ups? In: *Forensic Linguistics*, 4 (1997) 148–165.
6. Schmidt-Nielson, A. and Crystal, T.: Speaker Verification by Human Listeners: Experiments Comparing Human and Machine Performance Using the NIST 1998 Speaker Evaluation Data In: *Digital Signal Processing*, Vol. 10, Nos. 1 – 3, (2000) 249–266.

Speaker Discriminative Weighting Method for VQ-Based Speaker Identification

Tomi Kinnunen and Pasi Franti

University of Joensuu, Department of Computer Science
P.O. Box 111, 80101 Joensuu, Finland
{tkinnu, franti}@cs.joensuu.fi

Abstract. We consider the matching function in vector quantization based speaker identification system. The model of a speaker is a codebook generated from the set of feature vectors from the speaker's voice sample. The matching is performed by evaluating the similarity of the unknown speaker and the models in the database. In this paper, we propose to use weighted matching method that takes into account the correlations between the known models in the database. Larger weights are assigned to vectors that have high discriminating power between the speakers and vice versa. Experiments show that the new method provides significantly higher identification accuracy and it can detect the correct speaker from shorter speech samples more reliably than the unweighted matching method.

1 Introduction

Various phonetic studies have showed that different parts of speech signal have unequal discrimination properties between speakers. That is, the inter-speaker variation of certain phonemes are clearly different from other phonemes. Therefore, it would be useful to take this knowledge into account when designing speaker recognition systems.

There are several alternative approaches to utilize the above phenomenon. One approach is to use a front-end pre-classifier that would automatically recognize the acoustic units and give a higher significance for units that have better discriminating power. Another approach is to use weighting method in the front-end processing. This is usually realized by a method called *cepstral liftering*, which has been applied both in the speaker [3,9] and speech recognition [1]. However, all front-end weighting strategies depend on the parametrization (vectorization) of the speech and, therefore, do not provide a general solution to the speaker identification problem.

In this paper, we propose a new weighted matching method to be used in vector quantization (VQ) based speaker recognition. The matching takes into account the correlations between the known models and assigns larger weights for code vectors that have high discriminating power. The method does not require any *a priori* knowledge about the nature of the feature vectors, or any phonetic knowledge about the discrimination powers of the different phonemes. Instead, the method adapts to the statistical properties of the feature vectors in the given database.

2 Vector Quantization in Speaker Recognition

In VQ-based recognition system [4, 5, 6, 8], a speaker is modeled as a set of feature vectors generated from his/her voice sample. The speaker models are constructed by clustering the feature vectors in K separate clusters. Each cluster is then represented by a *code vector*, which is the centroid (average vector) of the cluster. The resulting set of code vectors is called a *codebook*, and it is stored in the speaker database.

In the codebook, each vector represents a single acoustic unit typical for the particular speaker. Thus, the distribution of the feature vectors is represented by a smaller set of sample vectors with similar distribution than the full set of feature vectors of the speaker model. The codebook should be set reasonably high since the previous results indicate that the matching performance improves with the size of the codebook [5, 7, 8]. For the clustering we use the *randomized local search* (RLS) algorithm as described in [2].

The matching of an unknown speaker is then performed by measuring the similarity/dissimilarity between the feature vectors of the unknown speaker to the models (codebooks) of the known speakers in the database. Denote the sequence of feature vectors extracted from the unknown speaker as $X = \{x_1, \dots, x_T\}$. The goal is to find the best matching codebook C_{best} from the database of N codebooks $C = \{C_1, \dots, C_N\}$. The matching is usually evaluated by a *distortion measure*, or *dissimilarity measure* that calculates the average distance of the mapping $d: X \times C \rightarrow \mathbf{R}$ [5, 8]. The best matching codebook can then be defined by the codebook that *minimizes* the dissimilarity measure.

Instead of the previous approaches, we use a *similarity measure*. In this way, we can define the weighting matching method intuitively more clearly. Thus, the best matching codebook is now defined as the codebook that *maximizes* the similarity measure of the mapping $s: X \times C \rightarrow \mathbf{R}$, i.e.:

$$C_{\text{best}} = \arg \max_{1 \leq i \leq N} \{s(X, C_i)\} \cdot \quad (2.1)$$

Here the similarity measure is defined as the average of the inverse distance values:

$$s(X, C_i) = \frac{1}{T} \sum_{t=1}^T \frac{1}{d(\mathbf{x}_t, \mathbf{c}_{\min}^{i,t})}, \quad (2.2)$$

where $\mathbf{c}_{\min}^{i,t}$ denotes the nearest code vector to \mathbf{x}_t in the codebook C_i and $d: \mathbf{R}^p \times \mathbf{R}^p \rightarrow \mathbf{R}$ is a given distance function in the feature space, whose selection depends of the properties of the feature vectors. If the distance function d satisfies $0 < d < \infty$, then s is a well-defined and $0 < s < \infty$. In the rest of the paper, we use *Euclidean distance* for simplicity. Note that in practice, we limit the distance values to the range $1 < d < \infty$ and, thus, the effective values of the similarity measure are $0 < s < 1$.

3 Speaker Discriminative Matching

Consider the example shown in Fig. 1, in which the code vectors of three different speakers are marked by rectangles, circles and triangles. There is also a set of vectors from an unknown speaker marked by stars. The region at the top rightmost corner cannot distinct the speakers from each other since it contains code vectors from all speakers. The region at the top leftmost corner is somewhat better in this sense because samples there indicate that the unknown speaker is not triangle. The rest of the code vectors, on the other hand, have much higher discrimination power because they are isolated from the other code vectors.

Let us consider the unknown speaker `star`, whose sample vectors are concentrated mainly around three clusters. One cluster is at the top rightmost corner and it cannot distinct, which speaker the sample vectors originate from. The second cluster at the top leftmost corner can rule out the speaker `triangle` but only the third cluster makes the difference. The cluster at the right middle indicates only to the speaker `rectangular` and, therefore, we can conclude that the sample vectors of the unknown speaker originate from the speaker `rectangular`.

The situation is not so evident if we use the unweighted similarity score of the formula (2.2). It gives equal weight to all sample vectors despite the fact that they do not have the same significance in the matching. Instead, the similarity value should depend on two separate factors: the distance to the nearest code vector, and the discrimination power of the code vector. Outliers and noise vectors that do not match well to any code vector should have small impact, but also vectors that match to code vectors of many speakers should have smaller impact on the matching score.

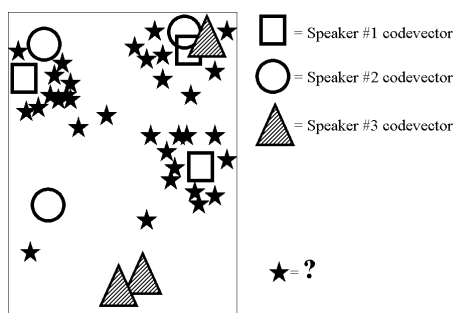


Fig. 1: Illustration of code vectors having different discriminating power.

3.1 Weighted Similarity Measure

Our approach is to assign weights to the code vectors according to their discrimination power. In general, the weighting scheme can be formulated by modifying the formula (2.2) as follows:

$$s_w(X, C_i) = \frac{1}{T} \sum_{t=1}^T \frac{1}{d(\mathbf{x}_t, \mathbf{c}_{\min}^{i,t})} \cdot w(\mathbf{c}_{\min}^{i,t}), \quad (3.1)$$

where w is the *weighting function*. When multiplying the local similarity score, $1/d(\mathbf{x}_t, \mathbf{c}_{\min}^{i,t})$, with the weight associated with the nearest code vector, $\mathbf{c}_{\min}^{i,t}$, the product can be thought as a local operator that moves the decision surface towards more significant code vectors.

3.2 Computing the Weights

Consider a database of speaker codebooks C_1, \dots, C_N . The codebooks are post-processed to assign weights for the code vectors, and the result of the process is a set of weighted codebooks $(C_i, W_i), i = 1, \dots, N$, where $W_i = \{w(\mathbf{c}_{i1}), \dots, w(\mathbf{c}_{iK})\}$ are the weights assigned for the i th codebook. In this way, the weighting approach does not increase the computational load of the matching process as it can be done in the training phase when creating the speaker database. The weights are computed using the following algorithm:

PROCEDURE ComputeWeights(S: SET OF CODEBOOKS) RETURNS WEIGHTS	
FOR EACH C_i IN S DO	% Loop over all codebooks
FOR EACH c_j IN C_i DO	% Loop over code vectors
sum := 0;	
FOR EACH $C_k, k \neq i$, IN S DO	% Find nearest code vector_
d _{min} := DistanceToNearest(c_j, C_k); % _ from all other codebooks	
sum := sum + 1/d _{min} ;	
ENDFOR	
w(c_j) := 1/sum;	
ENDFOR;	
ENDFOR;	

4 Experimental Results

For testing purposes, we collected a database of 25 speakers (14 males + 11 females) using sampling rate of 8.0 kHz with 16 bits/sample. The average duration of the training samples was 66.5 seconds per speaker. For matching

purposes we recorded another sentence of the length 8.85 seconds, which was further divided into three different subsequences of the lengths 8.85 s (100%), 1.77 s (20%) and 0.177 s (2%).

The feature extraction was performed using the following steps:

- High-emphasis filtering with filter $H(z) = 1 - 0.97z^{-1}$.
- 12th order mel-cepstral analysis with 30 ms Hamming window, shifted by 10 ms.

The feature vectors were composed of the 12 lowest mel-cepstral coefficients (except the 0th coefficient, which corresponds to the total energy of the frame). We concatenated the feature vectors also with the Δ - and $\Delta\Delta$ -coefficients (1st and 2nd time derivatives of the cepstral coefficients) to capture the dynamic behavior of the vocal tract. The dimension of the final feature vector is therefore $3 \times 12 = 36$.

The identification rates are summarized through Fig. 2-4 for the three different subsequences by varying the codebook sizes from $K=1$ to 256. The proposed method (weighted similarity) outperforms the reference method (unweighted similarity) in all cases. It reaches 100% identification rate with $K \geq 32$ using only 1.7 seconds of speech (corresponding to 172 test vectors). Even with a very short test sequence of 0.177 seconds (17 test vectors) the proposed method can reach identification rate of 84% whereas the reference method is practically useless.

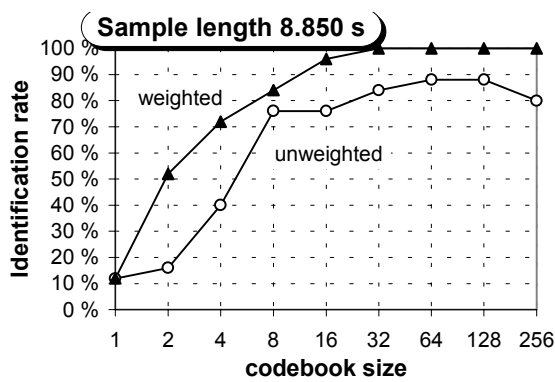


Fig. 2. Performance evaluation using the full test sequence.

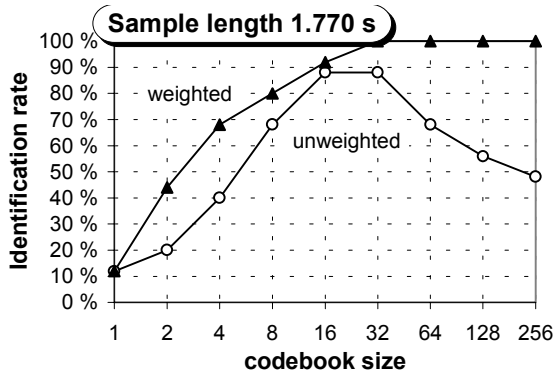


Fig. 3. Performance evaluation using 20 % of the test sequence.

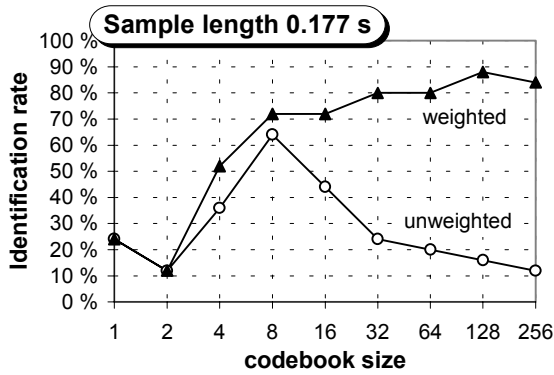


Fig. 4. Performance evaluation using 2 % of the test sequence.

5 Conclusions

We have proposed and evaluated a weighted matching method for text-independent speaker recognition. Experiments show that the method gives tremendous improvement over the reference method, and it can detect the correct speaker from much shorter speech samples. It is therefore well applicable in real-time systems. Furthermore, the method can be generalized to any other pattern recognition tasks because it is not designed for any particular features or distance metric.

References

- [1] Deller Jr. J.R., Hansen J.H.L., and Proakis J.G.: *Discrete-time Processing of Speech Signals*. Macmillan Publishing Company, New York, 2000.
- [2] Frnti P. and Kivijarvi J.: Randomized local search algorithm for the clustering problem, *Pattern Analysis and Applications*, **3**(4): 358-369, 2000.
- [3] Furui S.: Cepstral analysis technique for automatic speaker verification. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **29**(2): 254-272, 1981.
- [4] He J., Liu L., and Palm G.: A discriminative training algorithm for VQ-based speaker identification, *IEEE Transactions on Speech and Audio Processing*, **7**(3): 353-356, 1999.
- [5] Kinnunen T., Kilpelinen T., and Frnti P.: Comparison of clustering algorithms in speaker identification, *Proc. IASTED Int. Conf. Signal Processing and Communications (SPC)*: 222-227. Marbella, Spain, 2000.
- [6] Kyung Y.J. and Lee H.S.: Bootstrap and aggregating VQ classifier for speaker recognition. *Electronics Letters*, **35**(12): 973-974, 1999.
- [7] Pham T. and Wagner M.: Information based speaker identification, *Proc. Int. Conf. Pattern Recognition (ICPR)*, **3**: 282-285, Barcelona, Spain, 2000.
- [8] Soong F.K., Rosenberg A.E., Juang B-H., and Rabiner L.R.: A vector quantization approach to speaker recognition, *AT&T Technical Journal*, **66**: 14-26, 1987.
- [9] Zhen B., Wu X., Liu Z., and Chi H.: On the use of bandpass liftering in speaker recognition, *Proc. 6th Int. Conf. of Spoken Lang. Processing (ICSLP)*, Beijing, China, 2000.

Visual Speech: A Physiological or Behavioural Biometric?

J.D. Brand¹, J.S.D. Mason¹, and Sylvain Colomb²

¹ Department of Electrical Engineering
University of Wales Swansea, SA2 8PP, UK
J.D.Brand@swansea.ac.uk

² Ecole Nationale Supérieure de Physique de Marseille
Domaine Universitaire de Saint Jérôme
13013 Marseille, France

Abstract. This paper addresses an issue concerning the current classification of biometrics into either physiological or behavioural. We offer clarification on this issue and propose additional qualifications for a biometric to be classed as behavioural. It is observed that dynamics play a key role in the qualification of these terminologies. These are illustrated by practical experiments based around visual speech. Two sets of speaker recognition experiments are considered: the first uses lip profiles as both a physiological and a behavioural biometric, the second uses the inherent dynamics of visual speech to locate key facial features. Experimental results using short, consistent test and training segments from video recordings give recognition error rates as: physiological - lips 2% and face circles 11%; behavioural - lips 15% and voice 11%.

1 Introduction

In the definition given by Roethenbaugh [1] biometrics are “a measurable characteristic or trait of a human being for automatically recognising or verifying identity” The main goal of a biometric is to allow a specified user access to a closed system without the need for a physical key, ID card or password. Potential benefits of biometric technology stem from cost, accuracy and convenience: biometric systems can operate rapidly and with relatively low-cost technology; recent years have seen significant improvements in accuracy of biometrics generally. Furthermore, there is no requirement to carry or remember something like a key or PIN number. Of course all these parameters, benefits and disadvantages vary according to the type of biometric.

Highly accurate biometrics, such as retina scans and finger-prints, often possess negative attributes, such as above average expense and intrusiveness, requiring deliberate actions on behalf of the user. Other biometrics, can be cheaper, less intrusive and more covert. A good example of this is the particular form of automatic speaker verification proposed recently by Auckenthaler *et al.* [2] in the context of speaker verification. This system continuously monitors the speech signal entering a mobile phone. After automatically learning the characteristics

of the first person to use the device, it then continuously tracks usage, updating models as appropriate, and raises an alarm if usage does not include the said person for an excessive time. This is a good example of a fully covert, wholly non-intrusive biometric.

Table 1. A sliding scale for biometrics in terms of behavioural and physiological attributes.

	Physiological What you <i>Are</i>	Behavioural What you <i>Do</i>
Eye-related scan	are	
Fingerprints	are	
Speech (audio)		do
Handwriting		do
Gait	are →	do
Face	are →	do
Lips	are →	do

In 1998 Roethenbaugh and Mansfield revised a glossary of biometric terminology originally compiled by the Association of Biometrics (AfB) in 1993. Subsequently, a widely accepted classification of biometrics attempts to class systems as either physiological or behavioural. A physiological biometric is what you *are*, while a behavioural biometric is what you *do*. Thus a finger-print would be deemed a physiological biometric while a person’s gait or voice would be a behavioural biometric. In the limits DNA is the perfect physiological biometric having no behavioural information. A potential difficulty with these terms or concepts stems from the unavoidable link between the two classes. It is fairly obvious that behavioural biometrics imply movement or dynamics, and these dynamics will nearly always be dependent on the physiological make-up of the said person. One’s gait must be dependent on the physical properties of one’s legs! Some form of dependency will always hold for *all* behavioural biometrics.

Table 1 gives some examples of common biometrics, classified as either physiological or behavioural. Eye scans and fingerprints are clearly physiological with no obvious or widely used behavioural counterparts. Conversely, (audio) speech can be thought of only as a behavioural biometric, with what might be regarded as a weak link to physiological parameters. The arrows on the bottom three rows of Table 1 indicate that the physical links between the two classes is more obvious in the last cases; furthermore, in the cases of face and lips, both classes of biometric exist. Justification for the classification of *twin* biometrics stems from the fact that both are associated with biometric measurements which are both static and dynamic, as is explained in the next section.

Now as observed above, if there are no dynamics then there is no possibility of a behavioural biometric, but the converse is not true. From this it follows that problems of classification can arise when biometric measurements are made in the presence of dynamics. This paper offers clarification on this issue and proposes qualifications for a biometric to be classed as behavioural. It is argued that the role of *dynamics* to the principle of operation of the biometric is a key factor. In the following section the importance of dynamics to aid biometric classification is examined and illustrated using the case of visual speech. In particular the contours of lips are used to indicate the potential overlap between physiological and behavioural biometrics. It is the existence of dynamics which might well imply a behavioural biometric, but care is needed. Visual speech offers a good example of where confusion might arise.

Subsequent sections in the paper present experiments on two person recognition systems which illustrate the above point. Both use measurements recorded during speech, i.e. in a dynamic environment: the first uses lip profiles with static and dynamic properties; the second uses facial geometries and is classed as a physiological biometric, even though it uses dynamics to fully automate the biometric system.

As stated above physiological biometric can be categorised as something someone *has* or *is*, whereas a behavioural biometric is something someone *DOES* thereby implying movement or dynamics. Although this terminology is all encompassing, there is apparent overlap in that all behavioural biometrics are to a degree inextricably linked to physiological properties of the person in question i.e. everything we *do* stems from what we *are*. Visual speech provides an excellent vehicle to pursue this argument.

Figure 1 shows the components of visual speech along with an illustration of the speech production system. The acoustic signal, while obviously dependent on the vocal tract characteristics, is a clear example of a behavioural biometric. It is time-varying, and it is always so. Furthermore, without the relevant physical movement (of the diaphragm and vocal apparatus) the signal cannot exist; thus, nor can speaker verification! This suggests that one biometric qualifier might be: *without dynamics a behavioural biometric cannot exist*, where dynamics is defined as a significant or measurable level of pertinent movement over the time interval used to record a biometric 'signature'.

If the case for speech as a behavioural biometric is clear, the case for the lips is far less obvious. In the past, geometric lip profiles have been investigated for person recognition by a few researchers, including [6, 15]. The majority of work using lip profiles has been for speech reading [6, 7, 8, 9].

2 The Role of Dynamics in Biometrics

Lip profiles and the measurements thereof relate directly to the physical characteristics of a given person. Thus it would seem reasonable to classify the resultant biometric as physiological. But what if the lips are moving, during speech production for example? Does this then re-define the lip-based biometric as be-

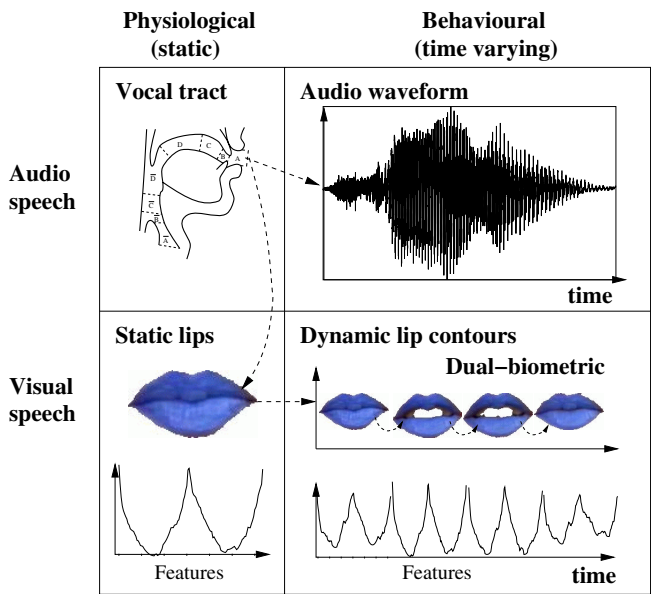


Fig. 1. Illustrating the inextricable link between behavioural biometrics and physiological components for audio and visual speech.

Table 2. The implications of dynamics on physiological and behavioural biometrics.

	Physiological	Behavioural
	<i>Has</i>	<i>Does</i>
Dynamics	Possible but unnecessary / detrimental	Essential
Biometric ‘signature’ variation	Possible / slow / small / nil	Inherent / unavoidable
System enrollment/training	Possible to be one-off	Multi-session / adaptive

havioural, or perhaps as a hybrid? We return to this question after reviewing the role of dynamics in biometrics.

Table 2 summarises the implications of dynamics in physiological and behavioural biometrics. Consider first the row entitled ‘Dynamics’. In the case of a physiological biometric, movement of any form is by definition unnecessary and might well be absent. In many such cases dynamics might well be undesirable, hampering the measurement process. This is a stark contrast therefore to the behavioural case where dynamics as defined above are essential. Note, it is all too easy at this stage to jump to the conclusion that if dynamics are present the biometric is behavioural and if not then it is physiological (though the latter is true).

Consider next the ‘Signature variation’ row in Table 2. The very existence of dynamics on whatever time scale causes unavoidable variations in the signatures. It is not possible to reproduce exactly the same speech signal, or exactly the same hand-written signature. Thus for physiological biometrics signature variations

can be very slow, small or even nil. Again this is in contrast to the behavioural case where signature variations are inevitable. This is one underlying reason why behavioural biometrics are often labelled as less accurate. In practice it is difficult if not impossible to capture all the natural variations exhibited in a behavioural biometric.

An important practical consequence of this observation is shown in the final row of Table 2. As a direct consequence of the variations inherent in behavioural biometrics, it is necessary to acquire larger quantities of training data to represent the person. In order to acquire this data it is usually necessary to have multiple enrolment sessions, or preferably an adaptive system that updates itself through usage [2], thereby minimising user inconvenience. In speaker recognition for example, Furui [10] and Thompson [11] have suggested that speech data should be collected over an interval of at least three months to capture typical variations. Any period of time shorter than this tends not to capture sufficient range of variations due to external influences such as ill health and mood fluctuations.

3 Physiological and Behavioural Speaker Recognition

Visual speaker recognition experiments are performed to highlight behavioural and physiological biometrics. The database comprises video recordings of 9 persons who each utter a series of prompted digits. In total each person utters 144 digits (12 digits per session for 12 sessions). In testing, an identification decision is made on each single digit and the error rates are averaged across the 9 speakers, 12 digits. Thus the experimental results come from a total of 1080 digit tests. Blue-highlighted lips are used throughout to make the task of automatic extraction possible. This way the lip features can be assessed in terms of their recognition potential instead of the extraction procedure. Both audio and visual recognition experiments are presented in [3, 12]. Here we consider just the visual case using a discrete cosine transform (ACT) of lip profiles as visual features.

The lip profiles, seen in Figure 2, represent a set or times series of *instantaneous* snap-shots, one set or 'signature' lasting for one spoken digit utterance. Movement can be measured from this time series by appropriate differentiation, resulting in dynamic features. The situation in the case of (audio) speech is different particularly in one important respect, namely the equivalent *instantaneous* snap-shots are not quite so instantaneous! In fact they come from a time window of speech recorded typically over 20ms or 30ms. Movement in the audio signal is inherent: it is a function of the the movement of the microphone. Dynamics from the time series of the lip profiles are measured by differentiating the time series of instantaneous features, leading to the visual behavioural biometric.

Table 2 shows visual speaker recognition error rates using instantaneous and dynamic features. The instantaneous visual features achieve better recognition error rates than the dynamic features. This is because of the inherent noise in the differentiating process to obtain these dynamic features.

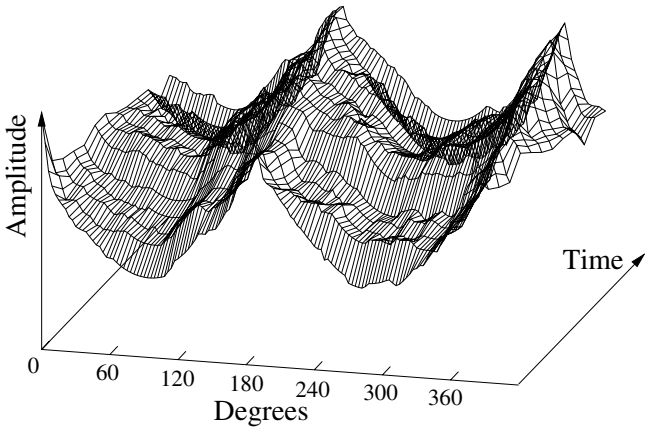


Fig. 2. A time series of lip profiles across one digit utterance.

Table 3. Visual speech: Instantaneous visual features are a physiological biometric, dynamic features are behavioural.

Feature	Instantaneous	Dynamic
Visual (DCT of lip profiles)	2%	15%

4 Dynamics to Locate Face Attributes

In this section visual dynamics associated with speech are utilised in a very simple yet fully automatic face recognition system based on simple geometric profiles, but accurately located and normalised.

Work on face analysis was performed as long ago as the 1970s [13], with key work twenty years later on PCA techniques by Kirby and Sirovich [14] and Turk and Pentland [15]. Subsequent studies have applied a wide variety of approaches, including various types of neural networks [16, 17], Hidden Markov Models (HMMs) [18] and shape analysis [19].

The components of a fully automatic face recognition system include: face detection to determine whether or not a face is present [19]; face location and segmentation to determine the boundaries of the face [20]; signal processing or feature extraction to derive representations suitable for similarity measures [21]; and a classifier or decision process. In a research environment, the first few components in the list are usually circumvented by the use of databases, invariably labelled by hand. Face extraction is also made simpler by using non-complex backgrounds. Here movement associated with normal speech is utilised to locate important facial attributes. During speech production, speakers naturally move their mouths, eyes and faces. These inherent dynamics are utilised to determine the:

- presence of a subject
- location of the mouth and eyes
- simple geometric normalised facial features.

The accurate determination of these items importantly provides a means of scaling the image to offset the differences in recording conditions, especially cross-session, and facilitates the direct use of facial geometry.

4.1 Eye and Mouth Location

Eye blinking has been used by Crowley *et al* [22,23] for multi-modal tracking of faces for video communications. People periodically blink their eyes to keep them moist, and when they speak they must naturally move their articulators. Frame differencing of an image sequence of a person speaking is used to locate mouth and eyes.

The dynamic information is obtained by comparing consecutive images along the time course. Here, two approaches are explored: first simple frame differencing and second normalised frame-to-frame correlation. Frame differencing is performed on a pixel by pixel basis, while normalised frame-to-frame correlation is performed on a series of square sub-images. Using frame differencing the motion of each pixel is given by the absolute difference in intensity between the corresponding pixels of two successive images. Normalised frame-to-frame correlation however takes into account image regions and gives the motion of the centre pixel in terms of a coefficient C_n given by:

$$C_n = \frac{\sum_{i,j} I_n(i,j) I_{n+1}(i,j)}{\sqrt{\sum_{i,j} I_n(i,j) \sum_{i,j} I_{n+1}(i,j)}} \quad (1)$$

where $I_n(i,j)$ is the pixel at position (i,j) of the sub-image n in the sequence. The denominator $\sqrt{\sum_{i,j} I_n(i,j) \sum_{i,j} I_{n+1}(i,j)}$ is a normalisation factor and is proportional to the energy in the appropriate regions of the images. Empirical experiments suggest that reasonable feature location can be achieved using a correlation sub-image of 9 pixels. Due to edge effects, the resultant image is therefore slightly smaller than the input images.

An average difference image is calculated over a digit utterance for both approaches. A typical spoken digit is between 0.5 and 1 second in duration, resulting in approximately 20 frames available for processing. The resultant image is termed a *dynamic face image*.

Figure 3 visually compares the two motion extraction methods and the resulting dynamic face images. It is found that the computationally more intensive normalised frame-to-frame correlation average achieves a more consistent feature location result, though the simple differencing may in practice be sufficient. Both dynamic face images also produce unwanted noise which surrounds the contours

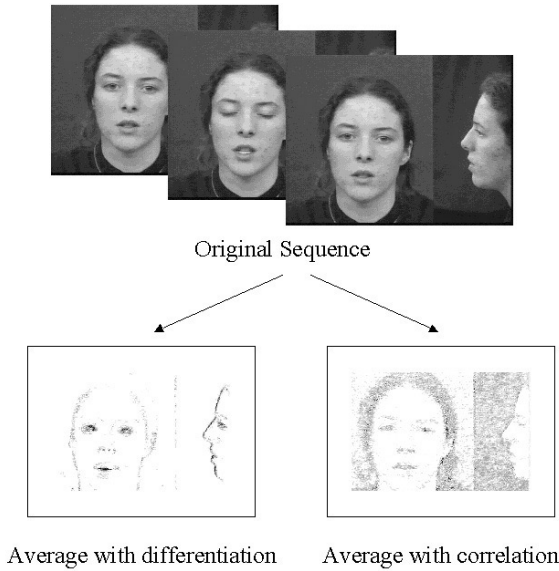


Fig. 3. Dynamic face images resulting from the average of the frame differencing sequence (left) and the average of the normalised frame-to-frame correlation sequence (right).

of the face. Therefore, an approach for filtering the differences is investigated. Instead of averaging a sequence of correlation frames, only the maximum value at each pixel is retained; thus each pixel location retains only its maximum value across an utterance. This is illustrated in the top right hand corner of Figure 4. Quantising this image to retain only the top 10% of pixel values gives a clear indication of eye and mouth locations, as illustrated by the bottom of Figure 5. Once the eye and mouth positions have been located, using standard blob extraction procedures [24], any future geometrical measurements can be normalised for scale using the eye to mouth ratio.

The simple facial profiles examined here are derived from concentric circles centred on the mouth. After performing histogram equalisation, topological grey level profiles as shown in Figure 5, are derived from the top half of the circles and normalised by the eye to mouth distance. Sobottka [25] uses similar topological profiles to locate facial features. Importantly, this accommodates cross-session variations of scale. The signatures produced by this procedure are tested for their speaker specific characteristics.

4.2 Experimental Conditions

A sub-section of the DAVID database [26] is used for the speaker recognition experiments. The subset comprises of 9 speakers uttering isolated English digits (12 in total). End-pointing is generous, thus utterances can be up to a second

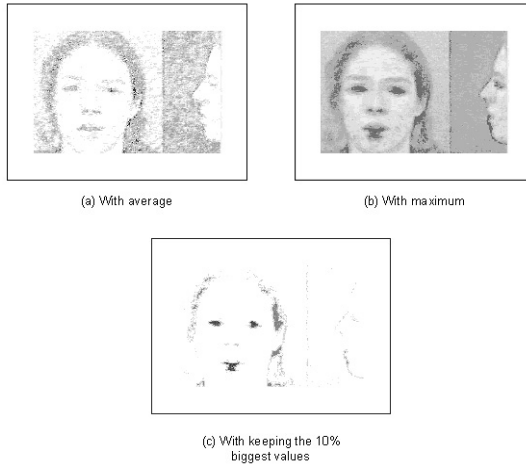


Fig. 4. Removing background noise from the normalised frame-to-frame correlation dynamic face images (top left) by taking the maximum pixel value from an image sequence (top right) and only keeping the top 10% (bottom).

in duration and are recorded at 30 frames per second. The background of the image sequences is non-complex.

There are 2 sets of data from 2 separate sessions, recorded on separate days. The position of the speaker is not rigidly constrained, resulting in a considerable variation in scale between sessions. Throughout all experiments the test token remains a single digit utterance from one session, with training on all digits from the opposite session. A round robin procedure is adopted, whereby the training and test sets are reversed. The average of the 2 results is then taken.

For comparison purposes audio speaker recognition experiments are also performed using the corresponding audio signal. Standard 14th order mel cepstra are used as audio features. The face profiles are matched by a one-dimensional non-linear alignment process.

4.3 Dynamic Face Image Results

Three profiles for each speaker are generated as illustrated by the bottom half of Figure 5. Only the top half of the circles are used, as the semi-circle below the lips often extend beyond the edges of the face and even beyond the edge of the image. The three profiles $R_{0.5}$, R_1 and $R_{1.5}$ correspond to ratios of eye-to-mouth distances. Empirical experiments suggest that the profile R_1 contains the most speaker discriminatory information achieving a recognition error rate of 11%. By coincidence the same score comes from audio recognition experiments. On first sight the audio experiments might be deemed to be poor. This can be explained by the extremely short training data (only 1 example per digit) under cross-session conditions. This emphasises the need to capture the natural variations in behavioural biometrics as mentioned above.

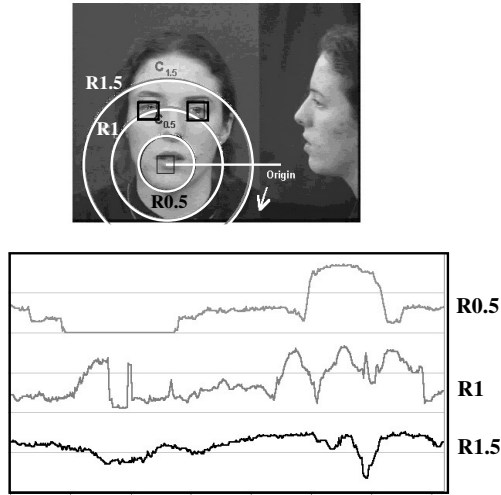


Fig. 5. Examples of circular face signatures where $R_{0.5}$, R_1 and $R_{1.5}$ correspond to ratios of eye-to-mouth distances.

5 Conclusion

This paper has addressed biometric classification. It is argued that the simple classification scheme of behavioural and physiological can be difficult to apply in cases like visual speech where there is potential for both physical and behavioural traits to be employed. It is observed that while dynamics are essential for behavioural biometrics to exist, it does not follow that the presence of dynamics means that a biometric is behavioural. The second experiment here illustrates this point: speech dynamics are used to locate facial geometries for a purely physiological biometric.

The first recognition experiments highlight the potential confusion in physiological and behavioural definitions. Lips have potential to be both. When they are stationary, then the biometric can be classed only as physiological; however, when instantaneous profiles are recorded during speech, then it is argued that the time series offers the opportunity to extract dynamic features, turning the biometric into a behavioural class, at least partially. For in this case it is impossible to de-couple the instantaneous component from the measurements. Experimental results using the same source of video clips and short, common test and training durations give recognition error rates as: physiological - lips 2% and face circles 11%; behavioural - lips 15% and voice 11%.

A very simple yet fully automatic approach to face recognition has also been developed, combining dynamic facial feature location with concentric topological grey level profile extraction. The process of feature detection capitalises on

the inherent motion of faces during the production of speech and is both an efficient and accurate means of feature location. It is also a method that can be easily adapted to other real-time face related applications, such as face detection and face-tracking. The performance of the visual face signatures is assessed via speaker recognition experiments. A baseline for comparison is obtained using the audio signal. Both audio and visual features achieve a recognition error rate of approximately 11% in cross-session conditions using limited quantities of data. Using more training data would undoubtedly improve the behavioural (audio) recognition scores.

In conclusion physiological biometrics are clear: fingerprints are a good example. However, behavioural biometrics often need qualifying: dynamics must exist and there must be a measurement in those dynamics influencing the biometric features. Even under these circumstances it seems difficult to conclude that any biometric can be purely behavioural, that is without any physiological influence in the biometric measure: the question is how much?

References

1. G. Roethenbaugh. Biometrics Explained. <http://www.icsa.net/services/consortia/cbdc/explained.shtml>, 1999.
2. R. Auckenthaler, E. Parris, and M. Carey. Improving a GMM Speaker Verification System by Phonetic Weighting. *ICASSP*, page 1440, 1999.
3. C. C. Chibelushi, J. S. Mason, and F. Deravi. Integration of acoustic and visual speech for speaker recognition. In *Proc. EuroSpeech*, volume 1, pages 157–160, Berlin, 1993.
4. P. Jourlin, J. Luetttin, D. Genoud, and H. Wassner. Acoustic Labial Speaker Verification. *Proc AVBPA, Lecture Notes in Computer Science 1206*, pages 319–334, 1997.
5. R. Auckenthaler, J. Brand, J. S. D. Mason, C. Chibelushi, and F. Deravi. Lip Signatures for Speaker Recognition. *AVBPA*, pages 142–147, Washington, 1999.
6. A. Rogozan and P. Deleglise. Continuous Visual Speech Recognition Using Geometric Lip-Shape Models and Neural Networks. *Proc 5th Conf. Speech Communication and Technology*, page 1999–2000, 1997.
7. I. Matthews, J. Bangham, R. Harvey, and S. Cox. Nonlinear Scale Decomposition Based Features for Visual Speech Recognition. *EUSIPCO98*, pages 303 – 305, 1998.
8. I. Matthews, J. Bangham, R. Harvey, and S. Cox. A Comparison of Active Shape Model and Scale Decomposition Based Features for Visual Speech Recognition. *ECCV*, pages 514–528, 1998.
9. R. Goecke, J. B. Millar, A. Zelinsky, and J. Robert-Ribes. Automatic Extraction of Lip Feature Points. In *Australian Conference on Robotics and Automation*, pages 31–36, 2000.
10. S. Furui. Speaker-Independent Isolated Word Recognition using Dynamic Features of speech spectrum. *IEEE Trans. on ASSP*, 34:52–59, 1986.
11. J. Thompson and J. S. Mason. Effects of Anxiety in Visual and Audio Speech Databases. *ESCA*, pages 21–24, 1995.
12. R. Auckenthaler, J. Brand, J. S. D. Mason, C. Chibelushi, and F. Deravi. Lip Signatures for Automatic Person Recognition. In *IEEE Workshop, MMSP*, pages 457–462, 1999.

13. T.Sakai, M.Nagao, and T.Kanade. Computer analysis and classification of photographs of human faces. *First U.S.A.-Japan Computer Conf. Proc.*, pages 55–62, 1972.
14. M. Kirby and L. Sirovich. Application of the Karhunen-Loève Procedure for the Characterization of Human Faces. *IEEE Trans. Pattern Analysis and Machine Intelligence*, pages 103–108, 1990.
15. M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, pages 98–113, 1991.
16. S. Lawrence, C. Giles, A. Tsoi, and A. Back. Face Recognition: A Convolutional Neural Network Approach. *IEEE Trans. Neural Networks*, pages 98–113, 1997.
17. H. Rowley, S. Baluja, and T. Kanade. Neural Network-based Face Detection. *IEEE Conf. on CVPR*, pages 203–207, 1996.
18. F. Samaria. Face Segmentation for Identification using Hidden Markov Models. In *Proceedings of 1993 British Machine Vision Conference*, pages 399–408, 1993.
19. A. Lantis, C.J. Taylor, and T.F. Cootes. An Automatic Face Identification System using Flexible Appearance Models. *Image and Vision Computing*, pages 393–401, 1995.
20. J. Hu, H. Yan, and M. Sakalli. Locating head and face boundaries for head-shoulder images. *PR*, 32:1317–1333, 1999.
21. A.Pentland and M.Turk. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, pages 71–86, 1991.
22. J.L. Crowley and F. Bérard. Multi-Modal Tracking of Faces for Video Communications. *IEEE Comp. Soc. Conf. CVPR*, pages 640–645, 1997.
23. Francois Bérard, Joëlle Coutaz, and James L. Crowley. Robust Computer Vision for Computer Mediated Communication. *INTERACT'97*, pages 581–582, 1997.
24. R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley Publishing Company, 1993.
25. K. Sobottka and Ioannis Pitas. Looking for Faces and Facial Features in Colour Images. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 1996.
26. C. C. Chibelushi, F. Deravi, and J. S. Mason. BT DAVID Database - Internal Report. *Speech and Image Processing Research Group, Dept. of Electrical and Electronic Engineering, University of Wales Swansea* URL: <http://www-ee.swan.ac.uk/SIPL/david/survey.html>, 1996.

An HMM-Based Subband Processing Approach to Speaker Identification

J.E. Higgins and R.I. Damper

Image, Speech and Intelligent Systems Research Group
Department of Electronics and Computer Science
University of Southampton, Southampton SO17 1BJ, UK
{jeh97r,rid}@ecs.soton.ac.uk

Abstract. This paper contributes to the growing literature confirming the effectiveness of subband processing for speaker recognition. Specifically, we investigate speaker identification from noisy test speech modelled using linear prediction and hidden Markov models (HMMs). After filtering the wideband signal into subbands, the output time trajectory of each is represented by 12 pseudo-cepstral coefficients which are used to train and test individual HMMs. During recognition, the HMM outputs are combined to produce an overall score for each test utterance. We find that, for particular numbers of filters, subband processing outperforms traditional wideband techniques.

1 Introduction

Automatic speaker recognition is an important, emerging technology with many potential applications in commerce and business, security, surveillance etc. Recent attention in speaker recognition has focused on the use of subband processing, whereby the wideband signal is preprocessed by a bank of bandpass filters to give a set of time-varying outputs, which are individually processed [2, 3]. Because these subband signals vary slowly relative to the wideband signal, the problem of representing them by some data model should be simplified [6].

The subband approach has also become popular in recent years in *speech* recognition [4, [15], [10]]. In this related area, the main motivation has been to achieve robust recognition in the face of noise. The key idea is that the recombination process allows the overall decision to be made taking into account any noise contaminating one or more of the partial bands. Hence, we investigate subband speaker identification in which narrowband noise is added to test utterances. The speech is modelled using linear prediction and hidden Markov models (HMMs).

The remainder of this paper is organised as follows. Section 2 describes subband processing and its possible benefits to an identification system. Section 3 briefly describes the speech database used and Section 4 details the feature extraction and data modelling processes. In Section 5, we describe the recombination of subband information and the decision rule used for the final identification. Section 6 gives results and Section 7 concludes.

2 Subband Processing

Figure 1 shows a schematic of the subband system used here. The bandpass filters are sixth-order Butterworth with infinite impulse response, designed using the bilinear transform. They are equally spaced on the mel-scale [14]. Filtering was performed in the time domain by direct calculation from the difference (recurrence) equation. Feature extraction is performed on each subband, and the resulting sequences of feature vectors are passed on to each subband's recognition algorithm. Thereafter, the outputs from each separate recogniser are fused, using multiple classifier techniques, to produce an overall decision as to the identity of the speaker.

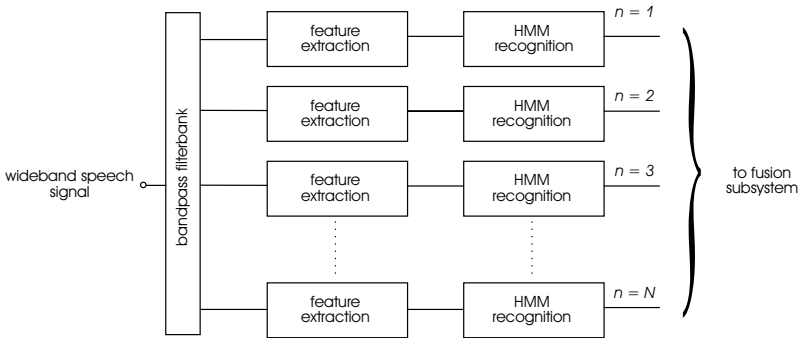


Fig. 1. Schematic diagram of the subband processing system. Each subband (filter) has its own recognition subsystem, whose output is fed to a fusion algorithm which makes the final, overall decision about speaker identity.

A successful recognition system is critically dependent on building good speaker models from the training data. In the system of Fig. 1 the problem arises at two points: extraction of features to represent the signal and building the recognition model. Data modelling, however, is subject to the well-known bias/variance dilemma [9]. According to this, models with too many adjustable parameters (relative to the amount of training data) will tend to overfit the data, exhibiting high variance, and so will generalise poorly. On the other hand, models with too few parameters will be over regularised, or biased, and will be incapable of fitting the inherent variability of the data. Subband processing offers a practical solution by replacing a large unconstrained data modelling problem by several smaller (and hence more constrained) problems [6].

3 Speech Database

In this work, we use the text-dependent British Telecom Millar database, specifically designed and recorded for text-dependent speaker recognition research. It consists of 46 male and 14 female native English speakers saying the digits *one* to *nine*, *zero*, *nought* and *oh* 25 times each. Recordings were made in 5 sessions spaced over 3 months, to capture the variation in speaker's voices over time which is one of the most important aspects of speaker recognition [7].

The speech was recorded in a quiet environment using a high-quality microphone, and a sampling rate of 20 kHz with 16-bit resolution. The speech data used here were downsampled to 8 kHz sampling rate, both to reduce the computation time necessary for our simulations and because this bandwidth is more typical of a real application. Data from the first two sessions (i.e., 10 repetitions of *seven*) were used for training and data from the remaining three sessions (15 repetitions) were used for testing.

As so far described, the speech data are essentially noise-free. However a major motivation behind subband processing has been the prospect of achieving good recognition performance in the presence of narrowband noise. Such noise affects the entire wideband model but only a small number of subbands. Hence, we have conducted identification tests with added noise. Following [3], Gaussian noise was filtered using a sixth-order Butterworth filter with centre frequency 987 Hz and bandwidth 365 Hz. It was added to the test tokens at a signal-to-noise ratio of 10 dB.

4 Data Modelling

In this work, we first have to model the speech signal. This is done by extracting features on a frame-by-frame basis. Many possible features could be extracted from the speech but here the feature set is based on cepstral coefficients. Cepstral analysis is motivated by, and designed for, problems centred on voiced speech [5]. It also works well for unvoiced sounds. Cepstral coefficients have been used extensively in speaker recognition [8], [13], mainly because a simple recursive relation exists that approximately transforms easily-obtained linear prediction coefficients into 'pseudo' cepstral ones [11]. The analysis frame was 20 ms long, Hamming windowed and overlapping by 50%. The first 12 coefficients were used (ignoring the zeroth cepstral coefficient, as usual).

Subsequently, we have to derive recognition models for the word *seven* spoken by the different speakers. For this, we use the popular hidden Markov models (HMMs). HMMs are powerful statistical models of sequential data that have been used extensively for many speech applications [12]. They consist of an underlying (hidden) stochastic process that can only be observed through a set of stochastic processes that produces an observation sequence. In the case of speech, this observation sequence is the series of feature vectors that have been extracted from an utterance (Section 4). Discrete HMMs were used with four states, plus a start and end state. Apart from self-loops (staying in the same

state), only left-to-right transitions are allowed. The frames of speech data were vector quantised and each HMM has its own linear codebook of size 32. Codebooks were constructed using a Euclidean distance metric. HMMs were trained and tested using the HTK software of: [16].

5 Score Combination and Decision Rule

[11] developed a common theoretical framework for combining classifiers which use distinct pattern representations. They outlined a number of possible combination schemes such as product, sum, min, max, and majority vote rules, and compared their performance empirically using two different pattern recognition problems. They found that the sum rule outperformed the other classifier combination schemes, in spite of theoretical assumptions apparently stronger than for the product rule. Further investigation indicated that the sum rule was the most resilient to estimation errors, which almost certainly explains its superior performance.

In this work, the HMM recognisers produce log probabilities as outputs. The use of logarithms is conventional, to avoid arithmetic underflow during computation. The fusion rule used here is that the identified speaker, i , is that for whom:

$$i = \arg \max_s [y^s] = \arg \max_s \sum_{n=1}^N \log p(\mathbf{x}|\omega(n, s)) \quad 1 \leq s \leq S$$

where N is the number of classifiers (subbands), and $p(\mathbf{x}|\omega(n, s))$ is the probability that model $\omega(n, s)$ for classifier n and model speaker s produced the observed data sequence \mathbf{x} , and y^s is the recombined (final) score for speaker s from the set of S speakers. Because of the use of logarithms, this is effectively the product rule but other rules tried worked no better.

6 Results

To test the subband system all 60 speakers in the database were used and the number of subbands was varied from 2 to 10. We compare the performance with the wideband (unfiltered) speaker identification system. The results are depicted in Figure 2.

Several interesting points can be gleaned from this figure. First, for six or more subbands, the subband system outperforms the wideband system. The best correct identification of 97.7% is obtained using 10 subbands, which was the maximum number used. The wideband system only achieves 65.2%. These results confirm the advantage of using a subband system in the face of narrowband noise. Second, using a small number of filters (< 6), subband performance is worse than the wideband system. The reason for this is currently unknown but is possibly because of the shape and location of the filters relative to the centre

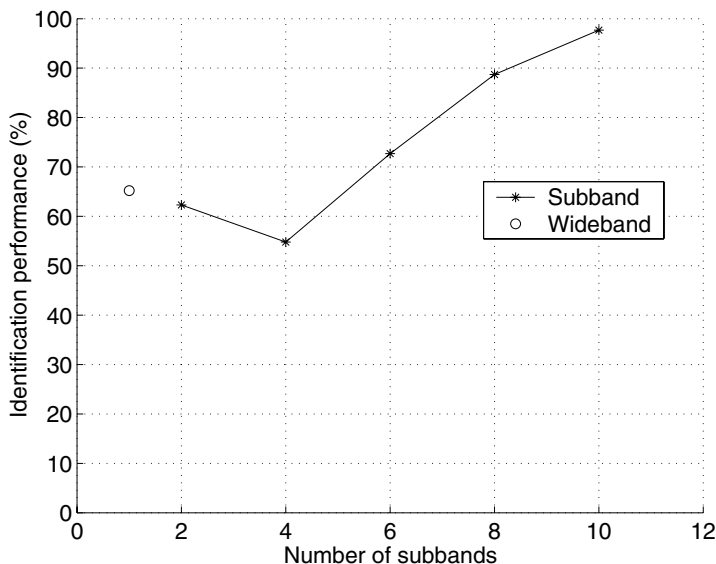


Fig. 2. Results for a 4-state HMM using 60 speakers, product fusion rule and various numbers of subbands. For comparison, the performance of a single, wide-band HMM recogniser is also shown.

frequency of the noise and/or those frequency regions which are important for discriminating speakers.

We are currently running further experiments in which larger numbers of subbands are being used as well as testing words other than just *seven*. These more complete results will be reported at the conference.

7 Conclusions

This paper contributes to the growing literature confirming the effectiveness of subband processing for speaker identification. The results confirm that subband processing offers improved performance (compared to the wideband system) in the face of narrowband noise. For the subband system tested here, ten subbands gave the best result. Future work will explore the use of more subbands and different words. We will also attempt to understand why the performance dips with four subbands.

Acknowledgements

The filter design program used here was written by Robert Finan. Author JEH is supported by a research studentship from the UK Engineering and Physical Science Research Council.

References

1. B.S. Atal, (1974). Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America* 55(6), 1304–1312.
2. L. Besacier and J.-F. Bonastre (1997). Subband approach for automatic speaker recognition: Optimal division of the frequency domain. In *Proceedings of 1st International Conference on Audio- and Visual-Based Biometric Person Authentication (AVBPA)*, Crans-Montana, Switzerland, pp. 195–202.
3. L. Besacier and J.-F. Bonastre (2000). Subband architecture for automatic speaker recognition. *Signal Processing* 80(7), 1245–1259.
4. H. Bourlard and S. Dupont (1996). A new ASR approach based on independent processing and recombination of partial frequency bands. In *Proceedings of Fourth International Conference on Spoken Language Processing, ICSLP'96*, Volume 1, Philadelphia, PA, pp. 426–429.
5. J.R. Deller, J.P. Proakis, and J.H.L. Hansen (1993). *Discrete-Time Processing of Speech Signals*. Englewood Cliffs, NJ: MacMillan.
6. R.A. Finan, R.I. Damper, and A.T. Sapeluk (2001). Text-dependent speaker recognition using sub-band processing. *International Journal of Speech Technology* 4(1), 45–62.
7. S. Furui, (1974). An analysis of long-term variation of feature parameters of speech and its application to talker recognition. *Electronic Communications* 57-A, 34–42.
8. S. Furui, (1981). Cepstral analysis techniques for automatic speaker verification. *IEEE Transactions on Acoustics, Speech and Signal Processing ASSP-29*(2), 254–272.
9. S. Geman, E. Bienenstock, and R. Doursat (1992). Neural networks and the bias/variance dilemma. *Neural Computation* 4(1), 1–58.
10. J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas (1998). On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(3), 226–239.
11. A. Morris, A. Hagen, and H. Bourlard (1999). The full-combination sub-bands approach to noise robust HMM/ANN-based ASR. In *Proceedings of 6th European Conference on Speech Communication and Technology, Eurospeech'99*, Volume 2, Budapest, Hungary, pp. 599–602.
12. L.R. Rabiner, (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77(2), 257–285.
13. D.A. Reynolds and R.C. Rose (1995). Robust text-independent speaker identification using Gaussian mixture models. *IEEE Transactions on Speech and Audio Processing* 3(1), 72–83.
14. S.S. Stevens and J. Volkman (1940). The relation of pitch to frequency: A revised scale. *American Journal of Psychology* 53(3), 329–353.
15. S. Tibrewala and H. Hermansky (1997). Sub-band based recognition of noisy speech. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'97*, Volume II, Munich, Germany, pp. 1255–1258.
16. S. Young, J. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland (2000). The HTK Book. Available from URL: <http://htk.eng.cam.ac.uk/>

Affine-Invariant Visual Features Contain Supplementary Information to Enhance Speech Recognition

Sabri Gurbuz, Eric Patterson, Zekeriya Tufekci, and John N. Gowdy

Department of Electrical and Computer Engineering
Clemson University, Clemson, SC 29634, USA
{sabbrig,epatter,ztufekc,jgowdy}@eng.clemson.edu

Abstract. The performance of audio-based speech recognition systems degrades severely when there is a mismatch between training and usage environments due to background noise. This degradation is due to a loss of ability to extract and distinguish important information from audio features. One of the emerging techniques for dealing with this problem is the addition of visual features in a multimodal recognition system. This paper presents an affine-invariant, multimodal speech recognition system and focuses on the supplementary information that is available from video features.

1 Introduction

A main difficulty of automatic speech recognition (ASR) systems is the loss of system performance that results from the addition of background noise to the environment in which the recognizer was trained. Rather than retraining a system for every possible environment, it is more desirable to be able to provide more sufficient information to a recognizer to overcome the degradation in system performance. An approach that has shown positive initial results for overcoming the negative effects of noise is the use of both audio and video features for speech recognition. It is well known that lipreading plays an important role in human speech understanding and the actual perception of what is spoken [1], [2]. The additional information available allows superior recognition results for both people and computers.

Lipreading is one of the most important facial characteristics, as it is directly related to the content of speech. Not only does the use of lip information mimic that of human perception, but it also allows use of information that may not be present in features taken from noise-degraded speech. Several have presented encouraging results using lipreading to supplement recognition information [3], [4], [5]. Here we present a robust visual feature extraction method that is affine (translation, rotation, scaling, and shear) invariant. Figure 1 shows the proposed late-integration-based joint audio-visual automatic speech recognition

(JAV-ASR) system. Separate audio and visual decisions are made and combined in the fusion algorithm. The decoding algorithm selects the model (such as a word or phoneme) that has the highest combined maximum-likelihood score. In our system, temporal affine-invariant Fourier descriptors of parametric outer-lip contours are used for video features that allow the speaker to move or rotate the head during speech. The video subsystem's decision is then combined with a standard, audio-only recognizer's decision. The combined results exceed those of either subsystem when the fusion is based on both the noise type and level tested for all eight noise types from NOISEX database.

This paper demonstrates the supplementary information with a proper weighting value between speech and video. This allows for superior speech recognition and could also be applied to other areas such as speaker recognition, and lip synchronization.

2 Joint Audio-Visual Speech Recognition System

This section describes the JAV-ASR system. The basic operation follows that of Figure 1 with independent audio and visual HMM-based subsystems, where the final decision is based upon a combination of the individual decisions. Figure 2 shows optimal ranges (shaded region) for the weighting value λ for STITEL, F16, Factory, and Car noises from the NOISEX database for several signal-to-noise ratios (similarly, optimal ranges for λ have been found for all NOISEX noises (data is based on SNRs of 18dB, 12dB, 6dB, 0dB, and -6dB or linearly 8:1, 4:1, 2:1, 1:1, 1:2). Each subsystem is described briefly in the following subsections.

2.1 Audio Subsystem

The audio subsystem is a standard HMM-based speech recognizer. Mel-frequency discrete wavelet coefficients are used for audio observation features. The discrete wavelet transform is applied to mel-scaled log-filterbank energies of speech frames. Isolated word recognition is implemented in HTK using left-to-right, eight-state HMMs. The complete description of the audio MFDWC feature extraction algorithm is described in [6].

2.2 Video Subsystem

The video subsystem is HMM-based, using dynamic affine-invariant Fourier descriptors (AI-FDs) for features. For the lipreading application, possible affine transformations on the lip contour data can be translation, scaling, rotation, and shear (uneven scaling of rotation matrix), alone or combined. The relationship between observed data \mathbf{x} and reference data \mathbf{x}^o can be written as,

$$\mathbf{x}[\mathbf{n}] = A\mathbf{x}^o[\mathbf{n} + \tau] + \mathbf{b}, \quad (1)$$

where A represents a 2×2 arbitrary matrix, $\det(A) \neq 0$, that may have scaling, rotation, and shearing affect, and \mathbf{b} represents a 2×1 arbitrary translation

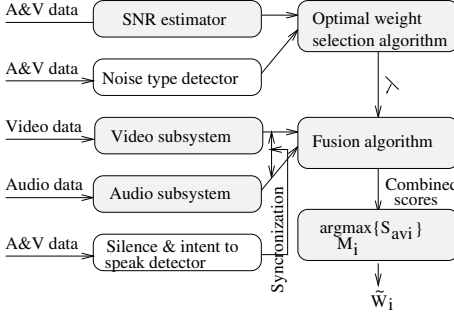


Fig. 1. Proposed JAV-ASR system.

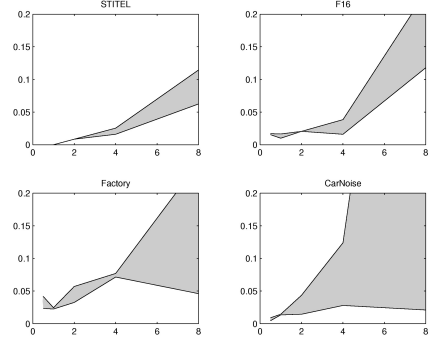


Fig. 2. λ versus linear SNR.

vector. Therefore we have a total of seven parameters to remove, which are four elements of A , two elements of \mathbf{b} , and the starting point \mathbf{p} . The algorithm extracts twelve AI-FDs of outer-lip contour data as well as four affine-invariant oral cavity features: width, height, ratio of width-to-height, and the area of the outer lip. Temporal coefficients are obtained by differencing the consecutive image sequence features. The complete description of the affine-invariant FDs algorithm is described in [7]. These dynamic coefficients are then used by the video HMM to generate log-likelihood scores [8].

Outer-Lip Contour Detection Algorithm. The basis of the video subsystem is an efficient outer-lip contour detection algorithm. Our technique uses color images with no prior labeling required. The goal is to segment lip regions and detect the outer lip contour. Then non-parametric data is fitted to an ellipsoid to find its parametric description [9], [7]. Figure 3 and 4 show the non-parametric and parametric outer-lip contour data superimposed on the mouth image, respectively. Here, $\mathbf{x} = [x \ y]^t$ is a vector representation of pixel locations on the contour.

3 Combining Audio and Video Decisions

This section briefly discusses the combination of the log-likelihood scores from the individual audio and video subsystems for an optimal joint-decision that exceeds the accuracy of either subsystem. Related work on this concept was published by other researchers in [10], [4]. A short description of late-integration follows. Let S_{ai} and S_{vi} log-likelihood scores of audio and video subsystems for the i^{th} HMM, respectively. Here, i is the index of the word, $1 \leq i \leq W$ (W is the vocabulary size), and M_i is the HMM for the i^{th} word. Using this notation, the combined likelihood score is computed using the following expression:

$$S_{avi} = \alpha S_{ai} + (1 - \alpha) S_{vi}, \quad i = 1, 2, \dots, W \quad (2)$$

where λ determines the influence of each subsystem upon final word decisions. When an optimal λ is found, the audio-visual recognition problem may be regarded as computing

$$\arg \max_{M_i} \{S_{avi}\}, i = 1, 2, \dots, W. \quad (3)$$

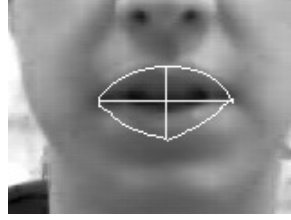
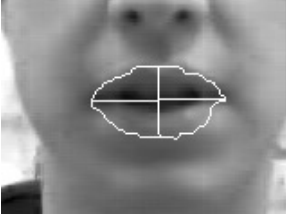


Fig. 3. Non-parametric lip contour data **Fig. 4.** Parametric lip contour data superimposed on top of the mouth image.

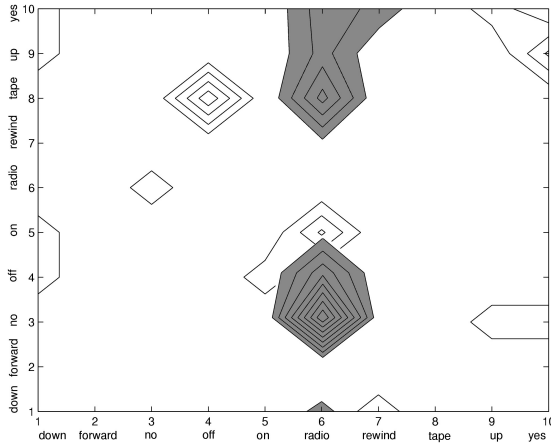


Fig. 5. Visual contour plot of confusion matrices. The mistakes made by the audio subsystem are shaded. The video are not.

4 Supplementary Information Provided by Visual Features

This section demonstrates the supplementary information available from visual features to enhance audio-only speech recognition decisions. Typical log-likelihood scores of audio, video and joint AV systems in a 10-word classification task where the speech signal was corrupted with factory noise at -6 dB are shown

in Table 1. For the Table 1, the test word was “**down**.” Table 1 shows that the audio subsystem ranks the word “**down**” fourth among all choices (an incorrect recognition), but the video subsystem ranks the word “**down**” first (correct recognition). Using the proper weighting value (see Figure 2), the JAV system properly recognizes the word “**down**”.

Table 2 shows the rank of correct classifications for each system for 1 test word and 17 test words from each of the 10 words where speech signal was corrupted with -6dB SNR factory noise. For the rank of one test word case in Table 2, the audio rank (AR) row, 2 out of 10 words are correctly ranked first. In the video rank (VR) row, 8 out of 10 words are correctly ranked first, but the audio-visual rank (AVR) row shows that nine out of ten words are correctly ranked first. For the average rank of 17 test words case in the same table, it is noticeable that the JAV system has better (lower) ranks consistently among the three systems. Table 3 shows the recognition accuracies of the audio, video and JAV systems for factory noise as well as other noises from NOISEX data base for various SNR values when λ is chosen based on training information for particular noises (see Figure 2 for the selection of λ for the SNR values and associated noise types seen in Table 3).

Table 1. Rank and typical log-likelihood scores of audio, video and joint AV systems in a 10-word classification task where the speech signal was corrupted with factory noise at -6 dB.

	audio subsystem		video subsystem		joint AV system	
Rank	word	log-likelihood sc.	word	log-likelihood sc.	word	log-likelihood sc.
1	tape	-1836.614868	down	712.612061	down	666.282827
2	off	-1847.899780	no	708.053467	no	660.214173
3	on	-1854.115479	tape	693.354248	tape	647.814806
4	down	-1861.234375	up	691.490540	up	642.723307
5	rewind	-1891.483643	yes	688.602417	on	640.599902
6	forward	-1907.797485	on	686.327881	yes	640.177119
7	no	-1949.685181	off	665.552368	off	620.310231
8	radio	-1964.797607	rewind	658.677185	rewind	612.774292
9	yes	-2001.692017	radio	654.391785	radio	607.246378
10	up	-2017.800293	forward	632.715210	forward	586.985983

Finally, Figure 4 presents a visual contour plot of the confusion matrices for all 17 test sets (170 words) for the audio subsystem (the shaded contours) and the video subsystem. (12dB-SNR audio was used with the audio subsystem because the recognition rates were closest to those of the video subsystem.) Contours are present and increase where mistakes are made by the recognizers. For example, the shaded contours indicate that the audio subsystem most often confused “no”, “off”, “tape”, “up”, and “yes” for the spoken word “radio,” whereas the unshaded contours indicate that the video subsystem made mistakes such as confusing “tape” for the spoken word “off.” It can easily be seen that the error regions of the audio subsystem and those of the video subsystem are

independent. These results all demonstrate that there is significant information provided to a speech recognition system with the addition of visual features.

Table 2. Rank of correct classification for each system for 1 test word and 17 test words from each word in the 10-word classification task where speech signal was corrupted with -6dB SNR factory noise. Abrivations: AR, VR and AVR represent audio, video and joint audio-visual systems’ ranks, respectively. Words: dw -down, rd -radio, and so on.

	Rank for one test word										Average rank for 17 test words									
	dw	fw	no	of	on	rd	rw	tp	up	ys	dw	fw	no	of	on	rd	rw	tp	up	ys
AR	4	8	9	1	3	1	2	2	2	6	2.7	4.9	8.2	2.2	2.5	2.5	1.7	3.3	2.5	6.2
VR	1	1	2	1	1	1	1	1	2	1	1.1	1	1.1	1.2	1.5	1.1	1	1.2	1.2	1.3
AVR	1	1	2	1	1	1	1	1	1	1	1	1	1.1	1.1	1.2	1	1	1.1	1.1	1.1

5 Experimental Setup

The JAV-ASR system is trained using audio and video features. We recorded the ten word spoken vocabulary at 30 fps a single speaker. Each word was recorded (audio and video) 17 times (total 170 utterances). To increase training and testing possibilities, 17 different training/test sets were created by rotating the 16 training sets among the 17 recordings, using the remaining set as the test set in each case.

Each subsystem individually generates log-likelihood scores for the ten word vocabulary. Finally, the fusion algorithm combines the decisions using λ chosen based on the type and level of the noise. The word with the highest score is then selected. Testing is done using the eight different noise types from the NOISEX data base (Table 3 shows only four noise cases) by corrupting clean speech with additive noise at various SNR levels. The audio subsystem was not trained with the noisy data, so the test system is unmatched.

6 Experimental Results

We are able to achieve 89.4% recognition accuracy with the visual information alone. The optimal decision algorithm incorporates an SNR and noise based weight selection rule that leads to a more accurate global likelihood ratio test. Table 3 show optimal λ values, which are chosen from Figure 2, the audio subsystem’s recognition accuracies, and the JAV-ASR system’s recognition accuracies for the various SNR values and noise types.

7 Conclusions and Future Work

This paper contributes the following: It validates the idea that the affine-invariant feature based video subsystem provides the supplementary information indepen-

dent of the audio subsystem that allows the JAV-ASR system to produce superior overall results. The supplementary information adds considerable robustness to natural speaking recognition systems and other systems such as person authentication. By basing the fusion rule on both the noise level and type, the combined audio-visual system outperformed either subsystem in all cases providing superior, noise-robust results. Future work includes testing with the addition of inner-lip contour information and extending the system to a large-vocabulary, continuous speech recognizer.

Table 3. Recognition accuracy of audio and joint AV systems (with video subsystem: 89.4%) when used with values of λ chosen from Figure 2. (optimal λ selection.)

	STITEL Noise			F16 Noise			Factory Noise			Car Noise		
	λ	A %	AV %	λ	A %	AV %	λ	A %	AV %	λ	A %	AV %
Clean	0.5154	100.0	100.0	0.5154	100.0	100.0	0.5154	100.0	100.0	0.5154	100.0	100.0
18 dB	0.0548	90.0	97.7	0.1654	96.5	100.0	0.0837	97.7	100.0	0.5129	100.0	100.0
12 dB	0.0144	65.3	93.5	0.0222	78.2	97.1	0.0172	83.5	97.7	0.0441	98.8	100.0
6 dB	0.0019	23.5	91.8	0.0028	37.1	94.1	0.0223	58.8	93.5	0.0239	88.8	98.2
0 dB	0.0007	15.9	90.6	0.0031	25.3	92.4	0.0050	44.7	92.9	0.0033	64.7	94.1
-6 dB	0.0004	11.2	90.0	0.0029	21.2	91.2	0.0180	24.7	92.9	0.0032	30.0	92.9

References

1. Massaro, D. W. and Stork, D. G.: Speech recognition and sensory integration. American Scientist, vol. 86, 1998.
2. Summerfield, Q.: Lipreading and audio-visual speech perception. Phil. Trans. R. Soc., vol. 335, 1992.
3. Petajan, E., Bischoff, B., Bodoff, D., and Brooke, N.: An improved automatic lipreading system to enhance speech recognition. in ACM SIGGHI, pp. 19-25, 1988.
4. Silsbee, P. L. and Bovik, A. C.: Computer lipreading for improved accuracy in automatic speech recognition. IEEE Transactions on Speech, and Audio Processing, vol. 4, no. 5, 1996.
5. Teissier, P., Robert-Ribes, J., Schwartz, J., and Guérin-Dugué, A.: Comparing models for audiovisual fusion in a noisy-vowel recognition task: IEEE Transactions on Speech, and Audio Processing, vol. 7, no. 6, 1999.
6. Gowdy, J. N. and Tufekci, Z.: Mel-scaled discrete wavelet coefficients for speech recognition. in Proceedings of ICASSP, 2000.
7. Gurbuz, S., Tufekci, Z., Patterson, E. K., and Gowdy, J. N.: Application of affine-invariant fourier descriptors to lipreading for audio-visual speech recognition. in Proceedings of ICASSP (accepted for publication), 2001.
8. Young, S., Odell, J., Ollason, D., Valtchev, V., and Woodland, P.: HTK Book. Cambridge University Press, 1997.
9. Wolfgang, S. and Schalkoff, R.J.: Direct surface parameter determination using an enhanced line-to-curve mapping approach. Optical engineering, vol. 36, no. 7, 1997.
10. Cox, S., Matthews, I., and Bangham, A.: Combining noise compensation with visual information in speech recognition. in Workshop on Audio-Visual Speech Processing (AVSP) pp 53-56, Rhodes, 1997.

Recent Advances in Fingerprint Verification

Anil K. Jain¹, Sharath Pankanti², Salil Prabhakar¹, and Arun Ross¹

¹ Dept. of Comp. Sci. and Eng., Michigan State University, East Lansing, MI 48824

² IBM T.J. Watson Research Center, Yorktown Heights, NY 10598

More than a century has passed since Alphonse Bertillon first conceived and then industriously practiced the idea of using body measurements for solving crimes [18]. Just as his idea was gaining popularity, it faded into relative obscurity by a far more significant and practical discovery of the uniqueness of the human fingerprints [1]. Soon after this discovery, many major law enforcement departments embraced the idea of first “booking” the fingerprints of criminals, so that their records are readily available and later using leftover fingerprint smudges (latents), they could determine the identity of criminals. These agencies sponsored a rigorous study of fingerprints, developed scientific methods for visual matching of fingerprints and strong programs/cultures for training fingerprint experts, and applied the art of fingerprint identification for nailing down the perpetrators [6].

Despite the ingenious methods improvised to increase the efficiency of the manual method of fingerprint indexing and search, the ever growing demands on manual fingerprint identification quickly became overwhelming. The manual method of fingerprint indexing resulted in a highly skewed distribution of fingerprints into bins (types): most fingerprints fell into a few bins and this resulted in search inefficiencies. Fingerprint training procedures were time-intensive and slow. Further, demands imposed by painstaking attention needed to visually match the fingerprints of varied qualities, tedium of monotonic nature of the work, and increasing workloads due to a higher demand on fingerprint identification services, all prompted the law enforcement agencies to initiate research into acquiring fingerprints through electronic medium and automatic fingerprint identification based on the digital representation of the fingerprints. These efforts have led to development of automatic/semi-automatic fingerprint identification systems (AFIS) over the past few decades.

While law enforcement agencies were the earliest adopters of the fingerprint identification technology, more recently, increasing identity fraud has created a growing need for biometric technology [9] for positive person identification in a number of non-forensic applications. Is this person authorized to enter this facility? Is this individual entitled to access the privileged information? Is the given service being administered exclusively to the enrolled users? Answers to questions such as these are valuable to business and government organizations. Since biometric identifiers cannot be easily misplaced, forged, or shared, they

¹ In 1893, the Home Ministry Office, UK, accepted that no two individuals have the same fingerprints.

² Biometric authentication, or simply biometrics, refers to use of distinctive physiological (e.g., fingerprints, face, retina, iris) and behavioral (e.g., gait, signature) characteristics for automatically identifying individuals.

are considered more reliable for personal identification than traditional token or knowledge based methods. The objectives of biometric authentication are user convenience (e.g., money withdrawal without ATM card and PIN), better security (e.g., difficult to forge access), and higher efficiency (e.g., lower overhead for computer password maintenance). Tremendous success of the fingerprint based identification technology in law enforcement applications, decreasing cost of the fingerprint sensing devices, increasing availability of inexpensive computing power, and growing identity fraud/theft have all ushered in an era of fingerprint-based person identification applications in commercial, civilian, and financial domains.

Our objective is to present current state-of-the-art in fingerprint sensing and identification technology and to provide some insights into the strengths and limitations of the automation in matching fingerprints. There is a popular misconception in the pattern recognition and image processing academic community that automatic fingerprint verification is a fully solved problem since it was one of the first applications of machine pattern recognition almost fifty years ago. On the contrary, fingerprint verification is still a challenging and important pattern recognition problem. Here, we will focus only on the core technology underlying fingerprint verification rather than the details of the commercial systems. In particular, we will discuss on fingerprint sensing, representation, classification, and matching. With the increase in the number of commercial systems for fingerprint-based verification, proper evaluation protocols are needed. The first fingerprint verification competition (FVC2000) was a good start in establishing such protocols. In order to improve the verification performance, methods for integrating multiple matchers, multiple biometrics and mosaicing of multiple templates are being investigated. As fingerprints (biometrics) get increasingly embedded into various systems (e.g., cellular phones), it becomes increasingly important to analyze the impact of biometrics on the overall integrity of the system and its social acceptability. We will also summarize some of the security/privacy research issues related to fingerprint (biometrics) authentication systems. A selection of fingerprint related research is cited below to provide the audience some useful pointers for their further exploration of this topic.

Bibliography

Books and Surveys

1. J. Cowger, *Friction Ridge Skin: Comparison and Identification of Fingerprints*. Elsevier, New York, 1983.
2. Federal Bureau of Investigation, *The Science of Fingerprints: Classification and Uses*. U.S. Government Printing Office, Washington, D. C., 1984.
3. National Institute of Standards and Technology, *Guideline for The Use of Advanced Authentication Technology Alternatives*. Federal Information Processing Standards Publication 190, 1994.
4. J. Rafferty and J. Wegstein, *The LX39 latent Fingerprint Matcher*. U.S.A. Government Publication. National Bureau of Standards, Institute for Computer Sciences and Technology, 1978.
5. D. R. Ashbaugh, *Quantitative-Qualitative Friction Ridge Analysis: An Introduction to Basic and Advanced Ridgeology*, CRC Press, Boca Raton, 1999.

6. H. C. Lee and R. E. Gaensslen (editors), *Advances in Fingerprint Technology*, Elsevier, New York, 1991.
7. F. Galton, *Finger Prints*, London: McMillan, 1892.
8. H. Cummins and Charles Midlo, *Fingerprints, Palms and Soles: An Introduction to Dermatoglyphics*. Dover Publications, Inc., New York, 1961.
9. A. K. Jain, R. M. Bolle, and S. Pankanti (editors), *Biometrics: Personal Identification in a Networked Society*, Kluwer Academic Publishers, 1999.
10. L. Hong, "Automatic Personal Identification Using Fingerprints", Ph. D. Thesis, Department of Computer Science and Engineering, Michigan State University, East Lansing, 1998.
11. S. Prabhakar, "Automatic Fingerprint Matching", Ph. D. Thesis, Department of Computer Science and Engineering, Michigan State University, East Lansing, 2001.
12. L. C. Jain, U. Halici, I. Hayashi, and S. B. Lee (eds.), *Intelligent Biometric Techniques in Fingerprint and Face Recognition*, CRC Press, Boca Raton, 1999.
13. C. Chapel. *Fingerprinting - A Manual of Identification*. Coward McCann, New York, 1971.
14. A. Moenssens. *Fingerprint Techniques*. Chilton Book Company, London, 1971.
15. A. K. Jain and S. Pankanti, "Fingerprint classification and matching", In A. Bovik, editor, *Handbook for Image and Video Processing*. Academic Press, April 2000.
16. A. K. Jain, L. Hong and S. Pankanti, "Biometrics Identification", Comm. ACM, pp. 91-98, Feb. 2000.
17. J. L. Wayman, "National Biometric Test Center: Collected Works 1997-2000", <http://www.engr.sjsu.edu/biometrics/nbtccw.pdf> 2000.
18. H. T. F. Rhodes. *Alphonse Bertillon: Father of Scientific Detection*. Abelard-Schuman, New York, 1956.
19. S. Pankanti, R. Bolle, and A. K. Jain (Guest Editors), Special Issue of the IEEE Computer Magazine on Biometrics, Feb 2000.
20. Proceedings of the First Audio and Video-Based Person Authentication, Crans-Montana, Switzerland, 12-14 March 1997.
21. Proceedings of the Second Audio and Video-Based Person Authentication, Washington D. C. USA, March 22-23, 1999
22. Proceedings of the IEEE, Special Issue on Biometrics, Vol. 85, No. 9, 1997.
23. B. Miller, "Vital signs of identity," IEEE Spectrum, Vol. 31, pp. 22-30, February 1994.
24. First Workshop on Automatic Identification Advanced Technologies, Stoney Brook, New York, United States, 1997.
25. Second Workshop on Automatic Identification Advanced Technologies, Morristown, New Jersey, United States, 1999.
26. Biometrics Consortium, www.biometrics.org
27. International Biometrics Industry Association, www.ibia.org.
28. BIOAPI, www.bioapi.org.

Fingerprint Scanning Devices

29. Digital Biometrics, Inc., biometric identification products. <http://www.digitalbiometrics.com>
30. Siemens ID Mouse. www.siemens.com
31. Fidelica Fingerprint Scanner. www.fidelica.com.
32. GemPlus, "Gemplus - Products, Services - Hardware - GemPC430", <http://www.gemplus.com/products/hardware/gempcTouch440.htm>.

33. Precise Biometrics, "Precise 100 A, SC", www.precisebiometrics.com.
34. Oxford Micro Devices Inc., "OMDI (finger) Imaging Technology that can help children", www.oxfordmicro.com.
35. Veritouch, "VR-3 (U)", www.veritouch.com.
36. Access Control Applications using Optical Computing. <http://www.mytec.com>.
37. Edge Lit Hologram for Live-scan Fingerprinting. <http://eastview.org/ImEdge>.
38. Scanner Specifications. <ftp://ard.fbi.gov/pub/IQS/spec>.
39. Thomson CSF. http://www.tcs.thomson-csf.com/fingerchip/FC_home.htm.
40. Y. Fumio, I. Seigo, and E. Shin, "Real-time fingerprint sensor using a hologram", *Applied Optics*, 31(11):1794, 1992.
41. J. Klett, "Thermal imaging fingerprint technology," In *Proc. Biometric Consortium Ninth Meeting*, Crystal City, Virginia, April 1997.
42. Lexington Technology, Inc. *Lexington Technology, Inc. Homepage*. <http://www.lexingtontech.com>.
43. I. Seigo, E. Shin, and S. Takashi, "Holographic fingerprint sensor," *Fujitsu Scientific and Technical Journal*, 25(4):287, 1989.
44. Harris Semiconductor. *Harris Semiconductor Homepage*. <http://www.semi.harris.com/fngrloc>.
45. TRS. *Technology Recognition Systems Homepage*. <http://www.betac.com/trs>.
46. W. Bicz, D. Banasiak, P. Bruciak, Z. Gumieny, S. Gumulinski, D. Kosz, A. Krysiak, W. Kuczynski, M. Pluta, and G. Rabiej, *Fingerprint structure imaging based on an ultrasound camera*. <http://www.optel.com.pl/article/english/article.htm>.
47. N. D. Young, G. Harkin, R. M. Bunn, D. J. McCulloch, R. W. Wilks, and A. G. Knapp, "Novel fingerprint scanning arrays using polysilicon tft's on glass and polymer substrates," *IEEE Electron Device Letters*, 18(1):19-20, Jan 1997.
48. Digital Persona, "Optical Fingerprint Sensors", www.digitalpersona.com.
49. G. V. Piosenka and R. V. Chandos, "Unforgeable personal identification system", U. S. Patent 4,993,068, 1991.
50. D. R. Setlak, "Fingerprint sensor having spoof reduction features and related methods", U.S. Patent 5,953,441, 1999.
51. Lapsley et al., "Anti-fraud biometric scanner that accurately detects blood flow", U.S. Patent 5,737,439, 1998.

Fingerprint Enhancement

52. S. Ghosal, N. K. Ratha, R. Udupa, and S. Pankanti, "Hierarchical partitioned least squares filter-bank for fingerprint enhancement," *15th IAPR International Conference on Pattern Recognition*, Barcelona, Spain, Sep. 3-7, pp. 334-337, 2000.
53. P. E. Danielsson and Q. Z. Ye, "Rotation-invariant operators applied to enhancement of fingerprints," In *Proc. 9th ICPR*, pages 329-333, Rome, 1988.
54. T. Kamei and M. Mizoguchi, "Image filter design for fingerprint enhancement," In *Proc. ISCV' 95*, pages 109-114, Coral Gables, FL, 1995.
55. E. Kaymaz and S. Mitra, "A novel approach to Fourier spectral enhancement of laser-luminescent fingerprint images," *Journal of Forensic Sciences*, 38(3):530, 1993.
56. A. Sherstinsky and R. Picard, "M-Lattice: From Morphogenesis to Image Processing," *IEEE Transactions on Image Processing*, Vol. 5, No. 7, pp. 1137-1150, July 1996.
57. D. C. D. Hung, "Enhancement and Feature Purification of Fingerprint Images," *Pattern Recognition*, vol. 26, no. 11, pp. 1,661-1,671, 1993.

58. L. Hong, Y. Wan, and A. K. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 20, No. 8, pp. 777-789, 1998.
59. L. O'Gorman and J. V. Nickerson, "An Approach to Fingerprint Filter Design", *Pattern Recognition*, Vol. 22, No. 1, 29-38, 1989.
60. Q. Xiao and H. Raafat, "Fingerprint Image Postprocessing: A Combined Statistical and Structural Approach," *Pattern Recognition*, vol. 24, no. 10, pp. 985-992, 1991.
61. S. Prabhakar, A. K. Jain, J. Wang, S. Pankanti, and R. Bolle, "Minutiae Verification and Classification for Fingerprint Matching", *Proc. 15th International Conference on Pattern Recognition (ICPR)*, Vol. I, pp. 25-29, Barcelona, September 3-8, 2000.
62. L. Berdan and R. Chiralo, "Adaptive digital enhancement of latent fingerprints," In *Proc. Int. Carnahan Conf. on Electronic Crime Countermeasures*, pages 131-135, University of Kentucky, Lexington, Kentucky, 1978.
63. K. Millard, D. Monro, and B. Sherlock, "Algorithm for enhancing fingerprint images," *Electronics Letters*, 28(18):1720, 1992.
64. D. Sherlock, D. M. Monro, and K. Millard, "Fingerprint enhancement by directional Fourier filtering," *IEE Proc. Vis. Image Signal Processing*, 141(2):87-94, 1994.

Fingerprint Classification

65. M. M. S. Chong, T. H. Ngee, L. Jun, and R. K. L. Gay, "Geometric framework for Fingerprint Classification," *Pattern Recognition*, Vol. 30, No. 9, pp. 1475-1488, 1997.
66. K. Goto, T. Minami, and O. Nakamura, "Fingerprint classification by directional distribution patterns," *System Computer Controls*, 13:81-89, 1982.
67. D. Maio and D. Maltoni, "A structural approach to fingerprint classification," In *Proc. 13th ICPR*, pages 578-585, Vienna, 1996.
68. A. K. Jain, S. Prabhakar, and L. Hong, "A Multichannel Approach to Fingerprint Classification", *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 21, No. 4, pp. 348-359, 1999.
69. A. P. Fitz and R. J. Green, "Fingerprint Classification Using Hexagonal Fast Fourier Transform," *Pattern Recognition*, Vol. 29, No. 10, pp. 1587-1597, 1996.
70. A. Senior, "A Hidden Markov Model Fingerprint Classifier," *Proceedings of the 31st Asilomar conference on Signals, Systems and Computers*, pp. 306-310, 1997.
71. B. G. Sherlock and D. M. Monro, "A Model for Interpreting Fingerprint Topology," *Pattern Recognition*, Vol. 26, No. 7, pp. 1047-1055, 1993.
72. B. Moayer and K. Fu, "A Tree System Approach for Fingerprint Pattern Recognition," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 8 no. 3, pp. 376-388, 1986.
73. C. L. Wilson, G. T. Candela, and C.I. Watson, "Neural Network Fingerprint Classification," *J. Artificial Neural Networks*, Vol. 1, No. 2, pp. 203-228, 1993.
74. C. V. Kameshwar Rao and K. Black, "Type Classification of Fingerprints: A Syntactic Approach," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 2, No. 3, pp. 223-231, 1980.
75. G. T. Candela, P. J. Grother, C. I. Watson, R. A. Wilkinson, and C. L. Wilson, "PCASYS: A Pattern-Level Classification Automation System for Fingerprints," *NIST Tech. Report NISTIR 5647*, August 1995.
76. K. Karu and A. K. Jain, "Fingerprint Classification," *Pattern Recognition*, Vol. 29, No. 3, pp. 389-404, 1996.

77. M. Kawagoe and A. Tojo, "Fingerprint Pattern Classification," *Pattern Recognition*, Vol. 17, No. 3, pp. 295-303, 1984.
78. R. Cappelli, D. Maio, and D. Maltoni, "Fingerprint Classification based on Multi-space KL", *Proc. Workshop on Automatic Identification Advances Technologies (AutoID'99)*, Summit (NJ), pp. 117-120, October 1999.
79. R. Cappelli, D. Maio, and D. Maltoni, "Combining Fingerprint Classifiers", *First International Workshop on Multiple Classifier Systems (MCS2000)*, Cagliari, pp.351-361, June 2000.
80. P. Baldi and Y. Chauvin, "Neural networks for fingerprint recognition," *Neural Computation*, 5(3):402-418, 1993.
81. B. Moayer and K. Fu, "An application of stochastic languages to fingerprint pattern recognition," *Pattern Recognition*, 8:173-179, 1976.
82. L. Hong and A. K. Jain, "Classification of Fingerprint Images," *11th Scandinavian Conference on Image Analysis*, June 7-11, Kangerlussuaq, Greenland, 1999.

Fingerprint Matching

83. R. Bahuguna, "Fingerprint verification using hologram matched filterings", In *Proc. Biometric Consortium Eighth Meeting*, San Jose, California, June 1996.
84. K. Balck and K. Rao, "A hybrid optical computer processing technique for fingerprint identification", *IEEE Trans. Computer*, 24:358-369, 1975.
85. B. Chatterjee and B. Mehtre, "Automatic fingerprint identification," *Journal of the Institution of Electronics and Telecom.*, 37(5/6):493, 1991.
86. M. Eleccion, "Automatic fingerprint identification," *IEEE Spectrum*, 10(9):36-45, 1973.
87. K. Fielding, J. Homer, and C. Makekau, "Optical fingerprint identification by binary joint transform correlation," *Optical Engineering*, 30:1958, 1991.
88. L. Frye, F. Gamble, and D. Grieser, "Real-time fingerprint verification system," *Applied Optics*, 31(5):652, 1992.
89. Q. Guisheng, C. Minde, Q. Shi, and N. Xue, "A new automated fingerprint identification system," *Computer Science Technology*, 4(4):289-294, 1989.
90. A. K. Hrechak and J. A. McHugh, "Automated Fingerprint Recognition Using Structural Matching," *Pattern Recognition*, Vol. 23, pp. 893-904, 1990.
91. A. K. Jain, L. Hong, S. Pankanti, and Ruud Bolle, "An identity authentication system using fingerprints," *Proceedings of the IEEE*, Vol. 85, No. 9, pp. 1365-1388, 1997.
92. A. K. Jain, L. Hong, and R. Bolle, "On-line Fingerprint Verification," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 19, No. 4, pp. 302-314, 1997.
93. A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based Fingerprint Matching," *IEEE Trans. Image Processing*, Vol. 9, No. 5, pp. 846-859, May 2000.
94. A. Ross, S. Prabhakar, and A. K. Jain, "Fingerprint Matching Using Minutiae and Texture Features", to appear in *International Conference on Image Processing (ICIP)*, Greece, October 7-10, 2001.
95. D. Maio and D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 19, No. 1, pp. 27-40, 1997.
96. L. Coetzee and E. C. Botha, "Fingerprint Recognition in Low Quality Images," *Pattern Recognition*, Vol. 26, No. 10, pp. 1141-1460, 1993.
97. L. O'Gorman, "Fingerprint Verification," in *Biometrics: Personal Identification in a Networked Society*, A. K. Jain, R. Bolle, and S. Pankanti (editors), Kluwer Academic Publishers, pp. 43-64, 1998.

98. N. Ratha, K. Karu, S. Chen, and A. K. Jain, "A Real-Time Matching System for Large fingerprint Databases," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 18, No. 8, pp. 799-813, 1996.
99. N. Ratha, S. Chen, and A. K. Jain, "Adaptive Flow Orientation-Based Feature Extraction in Fingerprint Images," *Pattern Recognition*, Vol. 28, No. 11, pp. 1657-1672, 1995.
100. S. Chen and A. K. Jain, "A Fingerprint Matching Algorithm Using Dynamic Programming", *Technical Report*, Department of Computer Science and Engineering, Michigan State University, 1999.
101. X. Quinghan and B. Zhaoqi, "An Approach to Fingerprint Identification by Using the Attributes of Feature Lines of Fingerprints," *Proc. Eighth Int. Conf. Pattern Recognition*, pp. 663-665, Oct. 1986.
102. C. Banner and R. Stock, "The FBI's approach to automatic fingerprint identification (part I)," *FBI Law Enforcement Bulletin*, U.S.A. Government Publication, 44(1), 1975.
103. C. Banner and R. Stock, "The FBI's approach to automatic fingerprint identification (part II)," *FBI Law Enforcement Bulletin*, U.S.A. Government Publication, 44(2), 1975.
104. I. Hideki, K. Ryuj, and H. Yu, "A fast automatic fingerprint identification method based on a weighted-mean of binary image," *IEICE Transactions on Fundamentals of Electronic*, 76:1469, 1993.
105. D. Isenor and S. Zaky, "Fingerprint identification using graph matching," *Pattern Recognition*, 19:113-122, 1986.
106. G. Johnson, D. McMahon, S. Teeter, and G. Whitney, "A hybrid optical computer processing technique for fingerprint identification," *IEEE Trans. Computers*, 24:358-369, 1975.
107. B. Mehtre, "Fingerprint image analysis for automatic identification," *Machine Vision and Applications*, 6(2-3):124-139, 1993.
108. A. Roddy and J. Stosz, "Fingerprint features - statistical analysis and system performance estimates," *Proceedings of IEEE*, 85(9):1390-1421, 1997.
109. M. Sparrow and P. Sparrow, *A Topological Approach to The Matching of Single Fingerprints: Development of Algorithms for Use on Latent Fingermarks*. U.S.A. Government Publication. Gaithersburg, MD: U.S. Dept. of Commerce, National Bureau of Standards, Washington, D.C., 1985.
110. E. Szekly and V Szekly, "Image recognition problems of fingerprint identification," *Microprocessors and Microsystems*, 17(4):215-218, 1993.
111. M. Trauring, "Automatic comparison of fingerprint-ridge patterns," *Nature*, 197(4871):938-940, 1963.
112. J. Wegstein, *The M40 Fingerprint Matcher*. U.S.A. Government Publication. Washington D.C.: National Bureau of Standards, Technical Note 878, U.S Government Printing Office, 1972.
113. J. H. Wegstein, *An Automated Fingerprint Identification System*. U.S.A. Government Publication, Washington, 1982.
114. R. Germain, A Califano, and S. Colville, "Fingerprint matching using transformation parameter clustering," *IEEE Computational Science and Engineering*, 4(4):42-49, 1997.

Performance Evaluation

115. A. K. Jain, S. Prabhakar, and A. Ross, "Fingerprint Matching: Data Acquisition and Performance Evaluation", *MSU Technical Report TR99-14*, 1999.

116. A. K. Jain, S. Prabhakar, and S. Pankanti, "Twin Test: On Discriminability of Fingerprints" to appear in *3rd International Conference on Audio- and Video-Based Person Authentication*, Sweden, June 6-8, 2001.
117. D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2000: Fingerprint Verification Competition", *Proc. 15th International Conference Pattern Recognition*, Barcelona, September 3-8, 2000, <http://bias.csr.unibo.it/fvc2000/>.
118. J. G. Daugman and G. O. Williams, "A Proposed Standard for Biometric Decidability," in *Proc. CardTech/SecureTech Conf.*, pp. 223-234, Atlanta, GA, 1996.
119. J. L. Wayman, "Multi-finger Penetration Rate and ROC Variability for Automatic Fingerprint Identification Systems", *Technical Report*, San Jose State University, 1999, <http://www.engr.sjsu.edu/biometrics/>.
120. J. L. Wayman, "Technical Testing and Evaluation of Biometric Identification Devices," In *Biometrics: Personal Identification in Networked Society*, Anil K. Jain, Ruud Bolle, and S. Pankanti (editors), Kluwer Academic publishers, pp. 345-368, 1998.
121. United Kingdom Biometric Working Group, "Best Practices in Testing and Reporting Biometric Device Performance", Version 1.0, March 2000. <http://www.afb.org.uk/bwg/bestprac10.pdf>.
122. Unpublished 1995 report by Frank Torpay of Mitre Corporation using data extracted from the FBI's Identification Division Automated Services database of 22,000,000 human-classified fingerprints.
123. R. Bolle, N. K. Ratha, and S. Pankanti, "Evaluating techniques for biometrics-based authentication systems", *Proc. 15th IAPR Int. Conference on Pattern Recognition*, Barcelona, Spain, Sep. 3-8, 2000.

Multimodal Fingerprint Systems

124. L. Hong, A. K. Jain and S. Pankanti, "Can Multibiometrics Improve Performance?", *Proceedings AutoID'99*, Summit, NJ, pp. 59-64, Oct 1999.
125. A. K. Jain, L. Hong, and Y. Kulkarni, "A Multimodal Biometric System using Fingerprint, Face, and Speech", *Proc. 2nd Int'l Conference on Audio- and Video-based Biometric Person Authentication*, Washington D.C., pp. 182-187, 1999.
126. A. K. Jain, S. Prabhakar, and S. Chen, "Combining Multiple Matchers for a High Security Fingerprint Verification System", *Pattern Recognition Letters*, Vol 20, No. 11-13, pp. 1371-1379, November 1999.
127. L. Hong and A. K. Jain, "Integrating Faces and Fingerprints For Personal Identification," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol.20, No.12, pp. 1295-1307, 1998.
128. S. Prabhakar and A. K. Jain, "Decision-level Fusion in Fingerprint Verification" to appear in *Pattern Recognition*, 2001.
129. U. Dieckmann, P. Plankensteiner, and T. Wagner, "Sesam: a Biometric Person Identification System Using Sensor Fusion," *Pattern Recognition Letters*, Vol. 18, No. 9, pp. 827-833, 1997.
130. J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On Combining Classifiers", *IEEE Trans. on Patt. Anal. and Machine Intell.*, Vol. 20, No. 3, pp. 226-239, 1998.
131. T. K. Ho, J. J. Hull, and S. N. Srihari, "On Multiple Classifier Systems for Pattern Recognition", *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 16, No. 1, pp. 66-75, 1994.

Miscellaneous

132. American National Standard for Information Systems, "Data Format for the Interchange of Fingerprint, Facial, and Scar Mark, & Tattoo (SMT) Information, Doc#ANSI/NIST-CSL ITL 1-2000, NIST Special Public Report 500-245", American National Standards Institute, New York, 2000
133. C. I. Watson and C. L. Wilson, "NIST Special Database 4, Fingerprint Database," National Institute of Standards and Technology, March 1992.
134. C. I. Watson and C. L. Wilson, "NIST Special Database 9, Fingerprint Database," National Institute of Standards and Technology, March 1992.
135. J. L. Wayman, "Daubert Hearing on Fingerprinting", http://www.engr.sjsu.edu/biometrics/publications_daubert.html.
136. R. Epstein and E. Hey, "Challenging the testimony of forensic identification experts in a post-daubert world: The new rules of the game", National Seminar for Federal Defenders, New Orleans, June 5-7, 2000.
137. J. Osterburg, T. Parthasarathy, T. E. S. Raghavan, and S. L. Sclove, "Development of a Mathematical Formula for the Calculation of Fingerprint Probabilities Based on Individual Characteristics", *Journal of the American Statistical Association*, No. 360, Vol. 72, 1997.
138. Problem Idents. <http://onin.com/fp/problemidents.html>
139. R. Cappelli, A. Erol, D. Maio, and D. Maltoni, "Synthetic Fingerprint-image Generation", Proc. International Conference on Pattern Recognition (ICPR), Barcelona, Vol. 3, pp. 475-478, September 2000.
140. F. Karen, "Encryption, smart cards, and fingerprint readers," *IEEE Spectrum*, 26(8):22, 1989.
141. D. Mintie, "Welfare id at the point of transaction using fingerprint and 2D bar codes," In *Proc. CardTech/SecurTech, Volume II: Applications*, pages 469-476, Atlanta, Georgia, May 1996.
142. C. Stanley, "Are fingerprints a genetic marker for handedness?" *Behavior Genetics*, 24(2):141, 1994.
143. C. M. Brislawn, *The FBI Fingerprint Image Compression Standard*. <http://www.c3.lanl.gov/~brislawn/FBI/FBI.html>.
144. R. M. Bolle, S. E. Colville, and S. Pankanti, "System and method for determining ridge counts in fingerprint image processing," U.S. Patent No. 6,111,978, 2000.
145. R. Cappelli, D. Maio, and D. Maltoni, "Modeling plastic distortion in fingerprints", *International Conference on advances in pattern Recognition (ICAPR2001)*, Rio Othon Palace Hotel, Rio de Janeiro, Brazil, March 11-14, 2001.
146. A. W. Senior and R. Bolle, "Improved Fingerprint Matching by Distortion Removal", *IEICE Transactions Special issue on Biometrics*, to appear, 2001.
147. M. Yeung and S. Pankanti, "Verification Watermarks on Fingerprint Recognition and Retrieval", *Journal of Electronic Imaging*, vol. 9, No. 4, pp. 468-476, October 2000.
148. N. K. Ratha, J. Connell, and R. Bolle, "Secure Biometric Authentication", *Proc. 1999 IEEE Workshop on Automatic Identification Advanced Technologies*, October 28-29, Morristown, NJ, 1999.
149. N. Ratha, J. Connell, and R. Bolle, "Cancelable Biometrics", *Biometrics Consortium Workshop*, September 2000.
150. J. Woodward, "Biometrics: Identifying law and policy concerns", in *Biometrics: Personal identification in a networked society*, A. K. Jain, R. Bolle, and S. Pankanti (eds.), Kluwer Academic Publishers, 1999.

Fast and Accurate Fingerprint Verification

Raghavendra Udupa U.¹, Gaurav Garg², and Pramod Sharma²

¹ IBM India Research Lab., Hauz Khas, New Delhi 100016, India

² Indian Institute of Technology, Hauz Khas, New Delhi 100016, India

(Extended Abstract)

Abstract. Speed and accuracy are the prerequisites of a biometric authentication system. However, most fingerprint verification methods compromise speed for accuracy or vice versa. In this paper we propose a novel fingerprint verification algorithm as a solution to this problem. The algorithm is inspired by a basic Computer Vision approach to model-based recognition of objects - *alignment*. We pose the problem of fingerprint matching as one of matching the corresponding feature sets. We propose a novel transformation consistency checking scheme to make verification accurate. We employ an early elimination strategy to eliminate inconsistent transformations and thereby achieve significant speed-up. Further speed-up is obtained by sampling based on geometric nearness. Our algorithm is simple, intuitive, easy to implement even on the simplest hardware and does not make any assumption on the availability of singularities like core and delta in the fingerprint. We report our results on three representative fingerprint databases.

1 Introduction

Reliable personal identification is critical in many applications in order to prevent identity fraud. Many applications demand real-time performance from the authentication systems. Unfortunately, fingerprint based personal identification systems proposed in literature sacrifice accuracy to speed or vice versa and make assumptions which do not hold always. In this paper we consider the problem of fast and accurate fingerprint verification and investigate how speed can be improved without sacrificing accuracy.

Fingerprint verification or *one-to-one* matching is the problem of confirming or denying a person's claimed identity by comparing a *claimant* fingerprint against an *enrollee* fingerprint. The presence of false minutiae, little common area between the fingerprints to be matched, sensory uncertainty, delocalization of minutiae, and elastic deformations make this problem very challenging. Previous attempts to solve the verification problem have modeled the problem as a graph matching problem [3], [8], as a syntactic pattern recognition problem [6], as a geometric matching problem [9], as an elastic string matching problem [4], and as an image matching problem [5].

In this paper, we propose a novel verification algorithm that employs a fundamental technique in Computer Vision-*alignment*. Our technique can handle

arbitrary amounts of translation and rotation and moderate elastic deformation. We pose the problem of matching two fingerprints as that of matching their minutiae sets. We assume that the two fingerprints are related approximately by a rigid transformation. We redefine notion of point-point match to address the problems caused by local non-linear elastic distortions, sensory uncertainty, and delocalization of minutiae during feature extraction. When the area common to two fingerprints varies significantly with prints, it is impossible to come up with a simple criterion for decision-making solely based on the number of point-point matches without affecting the error rates. We propose a novel transformation consistency checking scheme to handle this problem. The worst-case running time of the model-alignment algorithm is $O(m^3 n^2 \log n)$ and the worst-case behavior occurs whenever the two feature sets do not match [2]. We employ an early transformation elimination scheme to prune the search of transformation space while not hurting the accuracy of verification. We also employ an elegant geometric nearness based sampling technique to speed-up verification further. We report the results (accuracy and throughput) on three different databases.

1.1 Notations

The coordinates of a minutia p are denoted by $p.x$ and $p.y$ and the ridge angle is denoted by $p.\theta$. The ridge count of two minutiae p_1, p_2 is denoted by $R(p_1, p_2)$. The Euclidean distance between two minutiae p_1, p_2 is denoted by $D(p_1, p_2)$. $T(p)$ denotes the point to which p is mapped by transformation T .

1.2 Point-Point Matching

We say that two points p and q match under a rigid transformation T , if T takes p to a point which is in the close vicinity of q , i.e., $D(T(p), q) \leq \Delta_D$, where Δ_D is a small positive constant. Such a notion of point-point matching is necessary because of the inherent uncertainty associated with the geometric coordinates of minutiae. We can also compensate for local elastic deformations by choosing Δ_D appropriately. Similar notions of point-point matching have been reported in literature [9].

1.3 Transformations

We assume that the two minutiae set are related by a rigid transformation and compensate for elastic deformations by considering a tolerance box around each minutiae. Let (p_1, p_2) be a pair of minutiae in the *claimant* fingerprint and (q_1, q_2) be a pair in the *enrollee* fingerprint. When $D(p_1, p_2)$ and $D(q_1, q_2)$ differ significantly no rigid transformation T can map p_1 to q_1 and p_2 to q_2 such that $D(T(p_1), q_1) \leq \Delta_D$ and $D(T(p_2), q_2) \leq \Delta_D$. Hence, it makes sense to compute the transformation and explore it further only when $|D(p_1, p_2) - D(q_1, q_2)| \leq 2\Delta_D$. We could in fact, bin each minutiae pair (q_1, q_2) of the *enrollee* fingerprint based on $D(q_1, q_2)$. Then, for each minutiae pair (p_1, p_2) of the *claimant* fingerprint we can directly get all the potential matching pairs (q_1, q_2) . Once a

transformation T has been found out by hypothesizing a correspondence between (p_1, p_2) and (q_1, q_2) , T is applied on each minutia in the *claimant* fingerprint and the corresponding matching minutia of the *enrollee* fingerprint is determined if it exists. The result is a correspondence between a subset of the minutiae P' of the *claimant* fingerprint and a subset of the minutiae Q' of the *enrollee* fingerprint.

1.4 Transformation Consistency Checking

A simple cutoff based on the number of point-point matches very often results in false acceptances and false rejects especially when the number of minutiae is small. This is because, in many matching fingerprints the number of matching minutiae is small and those many accidental point-point matches are also possible in non-matching fingerprints. Our notion of point-point matching, which is very much necessary to handle elastics deformations, can often result in wrong pairings of minutiae. To make the matching process more robust we employ a transformation consistency checking scheme in which we determine the top K transformations (K is 10 in our implementation) in terms of number of point-point matches and check the consistency among these transformations. A rigid transformation T can be represented as a triplet (x, y, β) , where x and y are the translation along the X and Y axis respectively, and β is the rotation. We say that two transformations $T_1 = (x_1, y_1, \beta_1)$ and $T_2 = (x_2, y_2, \beta_2)$ are consistent if

1. $|x_1 - x_2| \leq \Delta_X$,
2. $|y_1 - y_2| \leq \Delta_Y$, and
3. $|\beta_1 - \beta_2| \leq \Delta_\beta$

where Δ_X , Δ_Y , and Δ_β are small positive constants. In case of matching fingerprints a majority of these transformations are mutually consistent while for non-matching fingerprints they are not. A score is computed based on the fraction of mutually consistent transformations.

1.5 Attributes Matching

A transformation establishes a geometrical correspondence between a subset P' of minutiae in the *claimant* fingerprint and a subset Q' of minutiae in the *enrollee* fingerprint. It needs to be further verified for topological correspondence. This involves determining the number of ridge angle matches and ridge count matches. Suppose minutia p of the claimant fingerprint matches with the minutia q of the enrollee fingerprint. This geometrical match is counted as a ridge angle match iff $|p.\theta + \beta - q.\theta| \leq \Delta_\theta$. If the geometrical match between pairs (p_1, p_2) and (q_1, q_2) also satisfies the inequality $|R(p_1, p_2) - R(q_1, q_2)| \leq \Delta_R$, it is regarded as a ridge count match. Here Δ_R and Δ_θ are small positive constants. The ridge angle and ridge count matches are appropriately normalized.

1.6 Decision Making

When the fraction of mutually consistent transformations is significant we evaluate the topological correspondence for each such transformation. A score is then

computed taking into account the percentage of point-point matches, the fraction of mutually consistent transformations, and the topological correspondence scores for each of the top transformations. This score is used to decide whether the *claimant* fingerprint matches with the *enrollee* fingerprint or not.

1.7 Speeding Up the Algorithm

The naive alignment algorithm is slow since it performs an exhaustive search of the transformation space defined by minutiae pair correspondences. For on-line applications, the high complexity of the naive algorithm is simply unacceptable. We now describe how the time complexity of the algorithm can be reduced significantly by applying some simple checks and by a novel sampling technique.

Early Elimination of Inconsistent Transformations. The attributes of minutiae (ridge angle, ridge count) can be effectively used to prune the search space as well as to make the matching more accurate. We employ some simple checks based on these attributes to eliminate inconsistent transformations. Let β be the rotational component of the transformation T mapping (p_1, p_2) to (q_1, q_2) .

1. If the pair (p_1, p_2) indeed matches with the pair (q_1, q_2) then the corresponding ridge counts must be nearly the same. Therefore, our first check is the following:

$$|R(p_1, p_2) - R(q_1, q_2)| \leq \Delta_R$$
2. The second test checks whether the ridge through each minutia is rotated by roughly the same amount

$$(|p_1.\theta + \beta - q_1.\theta| \leq \Delta_\theta)$$

$$(|p_2.\theta + \beta - q_2.\theta| \leq \Delta_\theta).$$

Here, Δ_R and Δ_θ are two small positive constants.

Sampling Based on Geometrical Nearness. An intuitive deterministic sampling technique is employed to speed up the verification algorithm. Recall that we use minutiae pairs to compute transformations. Instead of using every pair of minutiae, we use only those minutiae pairs whose separation falls in a pre-defined range $[d_{min}, d_{max}]$. Typically d_{min} is 50 and d_{max} is 150. The rationale behind using only minutiae separated by a distance in this region is as follows. The transformation obtained from points that are very close to each other are more susceptible to sensor error and elastic deformations while the probability of two minutiae that are far off occurring in a different print of the same finger is small. Sampling alone reduces the number of transformations to be tested from $mn(m-1)(n-1)/4$ to c^2mn where c is typically less than 10.

Overall Algorithm. Minutiae pairs of the enrollee and claimant fingerprints are sampled based on geometric nearness and binned. A match is hypothesized between a claimant minutiae pair (p_1, p_2) and an enrollee minutiae pair (q_1, q_2)

belonging to the same bin, and the corresponding rigid transformation T is computed. If the transformation is not consistent it is eliminated. Otherwise T is applied on claimant minutiae and the number of matches is determined. The process is repeated over other pairs and the set of $K(= 10)$ most promising transformations is determined. Transformation consistency among these transformations is determined. When the consistency is high, the number of ridge-angle matches and ridge-count matches is determined for each of these K transformations. A match score is computed based on the transformation consistency score and topological match.

2 Experimental Results

We tested our algorithm on three databases on a Pentium II 350 MHz machine. An in-house feature extractor was used for detecting minutiae. We report the FAR, FRR, and throughput (number of verifications/second) for these databases in Table 1. Database 1 consisted of a set of images (each of size 291×525) scanned by an optical fingerprint reader. There were 300 matching pairs and 19200 non-matching pairs. Database 2 consisted of a set of images (each of size 512×480) chosen randomly from the NIST 9 database. There were 900 matching pairs and 11683 non-matching pairs. Database 3 consisted of a set of images (each of size 508×480) scanned by a cheaper optical fingerprint reader. There were 1920 matching pairs and 10176 non-matching pairs.

Table 1. Error Rates and Throughput.

Database	FRR	FAR	Throughput (Non-matching)	Throughput (Matching)
1	1.33% (4/300)	0.00% (0/19200)	27	19
2	26.11% (235/900)	0.00% (0/11683)	1.5	1.1
	19.56% (176/900)	0.03% (3/11683)		
	14.44% (130/900)	0.23% (27/11683)		
3	5.56% (28/306)	0.00% (0/10176)	15	7
	4.25% (13/306)	0.15% (15/10176)		

The images in Database 1 and Database 3 had small to reasonable amount of elastic distortions. The images in Database 1 were in general of better quality than those in Database 3. Translation and rotation were significant (about 100-200 pixels and 10-30 degrees respectively) in many of the images. The images in

Database 1 had about 30-40 minutiae each on an average while those in Database 3 had about 40-60 minutiae. Database 2 consisted of very poor quality images with a significant amount of non-recoverable regions. In fact, for many mated pairs it would be difficult for a human expert to certify that they actually match! The feature extractor reported a high number of minutiae for these images (about 130) and many of the minutiae were false. The difference in the throughput for matching and non-matching fingerprints is because of the early elimination of inconsistent transformations. In case of non-matching fingerprints most of the transformations are inconsistent and are not explored further. In our experiments Δ_D was in the range 5-8, Δ_R in 2-3, Δ_θ in 5-10, d_{min} in 50-100, d_{max} in 100-200, Δ_X in 5-25, Δ_Y in 5-25, and Δ_β in 5-10.

3 Summary and Conclusions

We have described an algorithm for fast and accurate fingerprint verification. Our algorithm can handle arbitrary amounts of translation and rotation of the fingerprints. Our algorithm is very robust in the presence of noise and elastic deformations. Experimental results on representative databases are encouraging. A more detailed description of this work can be obtained from the first author.

References

1. D. Huttenlocher and S. Ullman, "Object Recognition using Alignment", *Proceedings of the 1st International Conference on Computer Vision*, pp. 102-111, 1987.
2. S. Irani and P. Raghavan, "Combinatorial and Experimental Results for Randomized Point Matching Algorithms", *Proceedings of the ACM Conference on Computational Geometry*, pp. 68-77, 1996.
3. D. K. Isenor and S. G. Zaky, "Fingerprint Identification using Graph Matching", *Pattern Recognition*, vol. 2, pp. 113-122, 1986.
4. A. K. Jain, L. Hong, and R. Bolle, "Online Fingerprint Verification", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 302-314, Apr. 1998,
5. A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "FingerCode: A Filterbank for Fingerprint Representation and Matching", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1999.
6. B. Moayer and K. S. Fu, "A Syntactic Approach to Fingerprint Pattern Recognition", *Pattern Recognition*, vol. 7, pp 1-23, 1975.
7. L. O'Gorman, "Fingerprint Verification", *Biometrics: Personal Identification in Networked Society*, Editors, A. K. Jain, R. Bolle and S. Pankanti, pp. 43-64, Kluwer Academic Publishers, 1999.
8. N. K. Ratha, V.D. Pandit, R. Bolle, and V. Vaish "High Accuracy Fingerprint Authentication using Elastic Subgraph Matching", Manuscript, 2000.
9. N. K. Ratha, K. Karu, S. Chen, and A. K. Jain, "A Real-Time Matching System for Large Fingerprint Databases", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 799-813. Aug. 1996,

An Intrinsic Coordinate System for Fingerprint Matching

Asker M. Bazen and Sabih H. Gerez

University of Twente, Department of Electrical Engineering
Laboratory of Signals and Systems
P.O. box 217 - 7500 AE Enschede - The Netherlands
Phone: +31 53 489 3156 Fax: +31 53 489 1060
`a.m.bazen@el.utwente.nl`

Abstract. In this paper, an intrinsic coordinate system is proposed for fingerprints. First the fingerprint is partitioned in regular regions, which are regions that contain no singular points. In each regular region, the intrinsic coordinate system is defined by the directional field. When using the intrinsic coordinates instead of pixel coordinates, minutiae are defined with respect to their position in the directional field. The resulting intrinsic minutiae coordinates can be used in a plastic distortion-invariant fingerprint matching algorithm. Plastic distortions, caused by pressing the 3-dimensional elastic fingerprint surface on a flat sensor, now deform the entire coordinate system, leaving the intrinsic minutiae coordinates unchanged. Therefore, matching algorithms with tighter tolerance margins can be applied to obtain better performance.

Keywords: Fingerprint matching, plastic distortions, flow lines, regular regions, intrinsic coordinate system.

1 Introduction

The first step in a fingerprint recognition system is to capture the print of a finger by a fingerprint sensor. In this capturing process, the 3-dimensional elastic surface of a finger is pressed on a flat sensor surface. This 3D-to-2D mapping of the finger skin introduces distortions, especially when forces are applied that are not orthogonal to the sensor surface. The effect is that the sets of *minutiae* (bifurcations and endpoints of the ridges) of two prints of the same finger no longer fit exactly. The ideal way to deal with distortions would be to invert the 3D-to-2D mapping and compare the minutiae positions in 3D. Unfortunately, there is no unique way of inverting this mapping.

Instead of modeling the distortions, most minutiae matching techniques use local similarity measures [11, 10], or allow some amount of displacement in the minutiae matching stage [1, 9]. However, decreasing the required amount of similarity not only tolerates small plastic distortions, but increases the *false acceptance rate* (FAR) as well. It is therefore reasonable to consider methods that explicitly attempt to model and eliminate the distortion. Such methods can be expected to be stricter in the allowed displacement during minutiae matching. As

a consequence, the FAR can be decreased without increasing the *false rejection rate* (FRR).

As far as we know, the only paper that addresses plastic distortions is [Cap01]. In that paper, the physical cause of the distortions is modeled by distinguishing three distinct concentric regions in a fingerprint. In the center region, no distortions are present, since this region tightly fits to the sensor. The outer, or external, region is not distorted either, since it does not touch the sensor. The outer region may be displaced and rotated with respect to the inner region, due to the application of forces while pressing the finger at the sensor. The region in between is distorted in order to fit both regions to each other.

Experiments have shown that this model provides an accurate description of the plastic distortions in some cases. The technique has successfully been applied to the generation of many synthetic fingerprints of the same finger [Cap00]. However, the model has not yet been used in an algorithm for matching fingerprints. Accurate estimation of the distortion parameters is still a topic of research. Furthermore, the prints cannot be truly normalized by the model, since it only describes relative distortions.

In this paper, an alternative approach is proposed, using a linked multilevel description of the fingerprint. The coarse level of this description is given by the *directional field* (DF). The DF describes the orientation of the local ridge-valley structures, thus modeling the basic shape of the fingerprint. We use a high-resolution DF estimate [Baz00] that is based on the averaging of squared gradient [Kas87]. The features at the detailed level of the fingerprint are given by the minutiae. Instead of the common practice of treating the DF and the minutiae as two separate descriptions, of which one is used for classification and the other for matching, we propose a link between these two levels, by defining the *intrinsic coordinate system* of a fingerprint.

The intrinsic coordinate system of a fingerprint is defined by the DF. One of its axes runs along the ridge-valley structures, while the other is perpendicular to them. Using the intrinsic coordinate system, minutiae positions can be defined with respect to positions in the DF, instead of using the pixel coordinates as minutiae locations, thus providing a more natural representation. If a fingerprint undergoes plastic distortions, the distortions do influence the shape of the coordinate system, but the intrinsic minutiae coordinates do not change. This means that matching the intrinsic coordinates of the minutiae sets is invariant to plastic distortions.

The rest of this paper is organized as follows. First, Section 2 proposes a method to partition the fingerprint in regular regions. Then, Section 3 defines the intrinsic coordinate system, which is used in Section 4 for a the minutiae matching algorithm. Finally, Section 5 presents some preliminary results.

2 Regular Regions

In this section, a method is proposed to partition the directional field in regular regions, which are the basis for the intrinsic coordinate system that is discussed in Section 3. The first step is extraction of the *singular points* (SPs). An SP is either a *core* or a *delta*, which are discontinuities in the DF. In [Baz01], a method is proposed for robust and accurate SP extraction from the high-resolution DF, based on computing the Poincaré index using small linear filters only. That paper also describes a method for the estimation of the *orientation* of the SPs by the convolution with a reference model of the SP. The orientation is used for initializing the flow lines.

The *flow lines*, which are traced in the DF, are used to partition the fingerprint into a number of regular regions. Flow lines are curves in a fingerprint that are exactly parallel to the ridge-valley structures. However, we also use this name for curves that are exactly perpendicular to the ridge-valley structures. Flow lines are found by taking line integrals in the directional field. This is discretized by a numerical Runge-Kutta integration, giving the following expression for tracing a flow line from a start position x_i which is a complex number that represents pixel coordinates in the fingerprint:

$$x_{i+1} = x_i + \Delta_x \cdot \text{mean}(DF_{x_i}, DF_{x_{i+1}}) \quad (1)$$

In this equation, Δ_x is the step size, DF_{x_i} is a unit-length complex number that indicates the orientation of the DF at x_i , $\text{mean}(DF_{x_i}, DF_{x_{i+1}}) = (DF_{x_i}^2 + DF_{x_{i+1}}^2)^{1/2}$ as discussed in [Baz01] and $DF_{x_{i+1}} = DF_{x_i + \Delta_x \cdot DF_{x_i}}$. For the perpendicular flow lines, steps that are perpendicular to the DF should be taken. This method of tracing flow lines gives relatively small errors and causes circular contours to be closed.

The extracted SPs and the flow lines are used to partition the fingerprints in so called *regular regions*, which are region in which no singular points are located. In order to construct the regular regions, two sets of flow lines have to be traced. From each *core*, a curve is traced that is parallel to the ridge-valley structure, and from each *delta*, three curves are traced that are perpendicular to the ridge-valley structure. This scheme provides a partitioning in regular regions for all classes of fingerprints and is illustrated in Figure 1(a) for a “right loop”. The most important property of regular regions is that the fingerprint in such a region can be warped non-linearly such that it only contains straight parallel ridge-valley structures. This is illustrated in Figure 2.

3 Intrinsic Coordinate System

The next step is to construct the *intrinsic coordinate system* (ICS) for each regular region in the fingerprint. This coordinate system is called intrinsic since it is the coordinate system that is defined by the DF of the fingerprint itself. Using the ICS, a multi-level description of the fingerprint can be made, in which the minutiae positions (fine level) are given with respect to their relative position

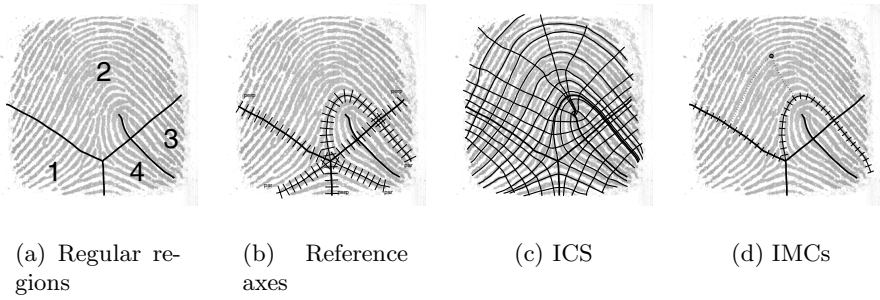


Fig. 1. Construction of the intrinsic coordinate system using flow lines.

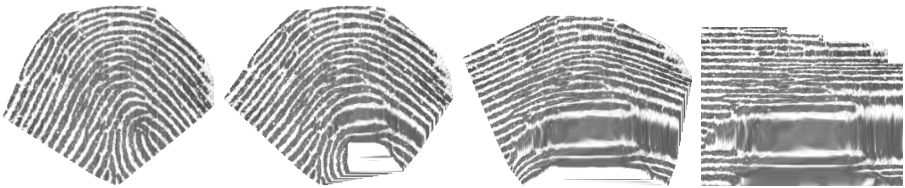


Fig. 2. Some intermediate stages in the non-linear warp of regular region 2.

in the DF (coarse level). Points in the fingerprint that are on the same ridge will have the same *perpendicular coordinate*, while all points on a curve that is perpendicular to the ridges share the same *parallel coordinate*.

The intrinsic coordinate system is defined in each regular region by two *reference axes*, which is illustrated in Figure 1(b). For each regular region, the first reference axis is given by the perpendicular curve through the delta, while the second reference axis is given by the parallel line through the delta. When there are no deltas in the fingerprint, there is only one regular region. In this case, any combination of one parallel and one perpendicular flow line can be taken as reference axes.

The resulting ICS grid can be visualized by tracing curves from equally spaced positions along the reference axes, as illustrated in Figure 1(c). Although the grid is equally spaced along the reference axes, this is not the case in the rest of the fingerprint. The parallel curves may for instance diverge because a ridge that is between them bifurcates.

The *intrinsic minutiae coordinates* (IMCs) can be determined directly from the DF by means of projection of the minutiae on the intrinsic axes. The parallel coordinate of a minutia is found by tracing a perpendicular curve until it crosses a parallel reference axis. The distance along the axis of this point to the origin of the ICS gives the parallel coordinate. The perpendicular coordinate of a minutia is found by tracing a parallel curve, until it crosses a perpendicular reference

axis. The distance along the axis of this point to the origin of the ICS gives the perpendicular coordinate. This method is illustrated in Figure 1(d).

4 Minutiae Matching in the ICS

A *minutiae matching* algorithm has to determine whether both fingerprints originate from the same finger by comparing the minutiae sets of a *primary* fingerprint (stored in a database) and a *secondary* fingerprint (provided for authentication). The goal of the algorithm is to find the mapping of the secondary to the primary minutiae set that maximizes the subset of corresponding minutiae in both sets. If the size of this common subset exceeds a certain threshold, the decision is made that the fingerprints are matching.

In this section, an alternative minutiae matching method that makes use of the ICS is proposed. As a consequence of the definition of the ICS, the IMCs only change in some simple and well-defined ways. Since distortions do not affect the ordering of the IMCs of neighboring minutiae, dealing with distortions amounts to independent 1-dimensional non-linear dynamic warps [Rah93] in 2 directions. This is a huge reduction in the number of degrees of freedom and computational complexity, compared to a 2-dimensional non-linear dynamic warp.

In the ICS, minutiae matching reduces to finding the warp function that maximizes the number of minutiae that fit exactly. Once the minutiae sets have been ordered along one intrinsic coordinate, the problem is to find the largest subset of both minutiae sets in which the ordering in the other intrinsic coordinate exactly corresponds. This problem can be solved by dynamic programming as described below. Usually, the next step would be to determine the warp function that interpolates the points that were found. However, since we are only interested in the number of matching minutiae, this step does not have to be performed.

The set of minutiae of the primary fingerprint is given by $\mathbf{a} = [a_1, \dots, a_m]$. A minutia a_i of this set is described by its parallel coordinate $x(a_i)$ and perpendicular coordinate $y(a_i)$. The set is sorted by $x(a_i)$. In the same way, \mathbf{b} describes the n minutiae of the secondary fingerprint. The problem is to find the longest series $[(a_{i_1}, b_{j_1}), \dots, (a_{i_k}, b_{j_k})]$ with $1 \leq i_1 < \dots < i_k \leq m$ and $1 \leq j_1 < \dots < j_k \leq n$ such that a criterion of local similarity is satisfied:

$$a_{i_k} - a_{i_{k-1}} \approx b_{i_k} - b_{i_{k-1}} \quad (2)$$

Consider the partial solution $\lambda(i, j)$, representing the longest series of minutiae pairs that has (a_i, b_j) as the last pair in the series. This means that $\lambda(i, j)$ can be constructed recursively by adding (a_i, b_j) to the longest of all series $\lambda(k, l)$, with $k < i$ and $l < j$, behind which the pair fits:

$$\lambda(i, j) = \lambda(k, l) + (a_i, b_j) \quad (3)$$

with k and l chosen to maximize the length of $\lambda(i, j)$. Using the starting conditions $\lambda(1, j) = [(a_1, b_j)]$ and $\lambda(i, 1) = [(a_i, b_1)]$, the final solution is the longest of all calculated partial solutions $\lambda(i, j)$.

5 Preliminary Results

Although the plastic distortion-invariant minutiae matching algorithm is not yet operational, this section is meant to show the potentials of our algorithm. In order to make the matching result only dependent on the ability to deal with plastic distortions, and not on the minutiae extraction algorithm, the minutiae were extracted by human inspection. After cross-validation of both sets, 77 corresponding minutiae were in the two prints of the same finger that are shown in Figure 3(a) and 3(b).

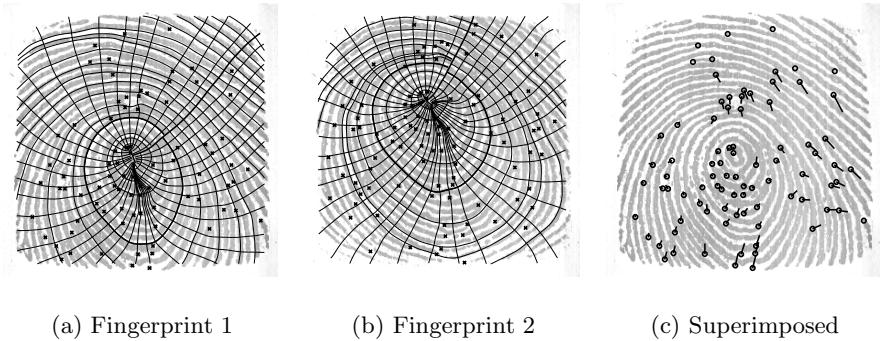


Fig. 3. Minutiae sets located in the ICSs of two fingerprints of the same finger.

In Figure 3(c), the sets have been aligned as well as possible, using only translation, rotation and scaling; matching minutiae are connected by a line. The figure clearly shows that, although the two minutiae sets fit exactly at the center of the print, they do not fit very well in the rest of the fingerprint. Under these distortions, the displacements required to tolerate corresponding minutiae is larger than the distance between some neighboring minutiae. Therefore, tolerance of small displacements is not a solution in this case. For a reasonable tolerance, only 24 matching minutiae pairs are found, out of 77 true matching minutiae pairs.

When the ICS is used, the perpendicular ordering is preserved since the parallel lines exactly follow the ridge-valley structures. However, the plastic distortions can influence the parallel ordering. Especially the parallel coordinate of minutiae near the core are less reliable. Nevertheless, over 70 correctly ordered minutiae pairs can be found using the ICS.

6 Conclusions

In this paper, the intrinsic coordinate system of a fingerprint is presented. It is defined by the directional field of the fingerprint itself. The locations of minutiae are insensitive to plastic distortions if they are given in this intrinsic coordinate system. It is shown how minutiae-based matching is drastically simplified when minutiae are characterized by means of their intrinsic coordinates.

References

- Baz00. A.M. Bazen and S.H. Gerez. Directional Field Computation for Fingerprints Based on the Principal Component Analysis of Local Gradients. In *Proceedings of ProRISC2000, 11th Annual Workshop on Circuits, Systems and Signal Processing*, Veldhoven, The Netherlands, November 2000.
- Baz01. A.M. Bazen and S.H. Gerez. Extraction of Singular Points from Directional Fields of Fingerprints. In *Mobile Communications in Perspective, CTIT Workshop on Mobile Communications, University of Twente*, pages 41–44, Enschede, The Netherlands, February 2001.
- Cap00. R. Cappelli, A. Erol, D. Maio, and D. Maltoni. Synthetic Fingerprint-Image Generation. In *Proceedings of ICPR2000, 15th Int. Conf. Pattern Recognition*, Barcelona, Spain, September 2000.
- Cap01. R. Cappelli, D. Maio, and D. Maltoni. Modelling Plastic Distortion in Fingerprint Images. In *proceedings of ICAPR2001, Second Int. Conf. Advances in Pattern Recognition*, Rio de Janeiro, March 2001.
- Jai97. A.K. Jain, L. Hong, S. Pankanti, and R. Bolle. An Identity-Authentication System Using Fingerprints. *Proc. of the IEEE*, 85(9):1365–1388, Sept. 1997.
- Jia00. X. Jiang and W.Y. Yau. Fingerprint Minutiae Matching Based on the Local and Global Structures. In *Proceedings of ICPR2000, 15th Int. Conf. Pattern Recognition*, volume 2, pages 1042–1045, Barcelona, Spain, September 2000.
- Kas87. M. Kass and A. Witkin. Analyzing Oriented Patterns. *Computer Vision, Graphics, and Image Processing*, 37(3):362–385, March 1987.
- Rab93. L. Rabiner and B.H. Juang. *Fundamentals of Speech Recognition*. Signal Processing Series. Prentice Hall, Englewood Cliffs, NJ, 1993.

A Triplet Based Approach for Indexing of Fingerprint Database for Identification

Bir Bhanu and Xuejun Tan

Center for Research in Intelligent Systems
University of California, Riverside, CA92521, USA
{bhanu,xtan}@cris.ucr.edu

Abstract. This paper presents a model-based approach which efficiently retrieves correct hypotheses using properties of triangles formed by the triplets of minutiae as the basic representation unit. We show that the uncertainty of minutiae locations associated with feature extraction and shear does not affect the angles of a triangle arbitrarily. Geometric constraints based on characteristics of minutiae are used to eliminate erroneous correspondences. We present an analysis to characterize the discriminating power of our indexing approach. Experimental results on fingerprint images of varying quality show that our approach efficiently narrows down the number of candidate hypotheses in the presence of translation, rotation, scale, shear, occlusion and clutter.

1 Introduction

Fingerprints have long been used for personal recognition. There are two kinds of fingerprint based biometric systems in terms of their utilization: verification and identification. In verification, the input includes a fingerprint image and an ID, the system then verifies whether the image is consistent with the ID. In identification, the input is only a fingerprint image and the system identifies fingerprint images in the database corresponding to the input fingerprint image. The problem of identifying a fingerprint image can be stated as: given a fingerprint database and a test fingerprint image, in the presence of translation, rotation, scale, shear, occlusion and clutter, does the test image resemble any of the fingerprints in the database? It is still a challenging problem. Recent techniques of fingerprint recognition are presented in [1]-[3]. All of them are for verification. Like our approach, Germain et al. [4] use the triplets of minutiae in the indexing procedure. However, the features they use are different from ours. The features they use are: the length of each side, the ridge counts between each pair, and the angles measured with respect to the fiducial side. The problems with their algorithm are: (a) the length changes are not insignificant under scale and shear; (b) the ridge count and the angle are both very sensitive to the quality of images. As a result, they have to use large size of bins to tolerate distortions, which reduces the size of index space and degrades the performance of their algorithm greatly. The key contributions of our work are to present a new indexing algorithm based on features derived from the triplets of minutiae and to demonstrate the power of it on a fingerprint database of 1000 images.

2 Triplet-Based Features for Indexing

The triangle features that we use are its angles, orientation, type, direction and maximum side.

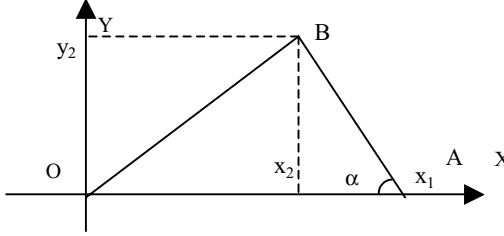


Fig. 1. Illustration of variables.

2.1 Analysis of Angle Changes under Distortions: It can be proved that angles are invariant under translation, rotation and scale. However, the transform between the different impressions of the same finger also includes the uncertainty of minutiae locations, which is associated with feature extraction and shear. Thus, the location of each minutia translates in a small local area randomly and independently.

Fig. 1 shows a triangle. Without loss of generality, we assume one vertex, O, of the triangle is (0, 0), and it does not change under distortions. Suppose the positions of points A and B are $(x_1, 0)$ and (x_2, y_2) , $x_1 > 0$, $y_2 > 0$ and $x_2 \in (-\infty, +\infty)$. Because of the uncertainty of locations, A and B move to $(x_1 + \Delta x_1, 0)$ and $(x_2 + \Delta x_2, y_2 + \Delta y_2)$, and α changes to $\alpha + \Delta\alpha$, respectively. Since for small $\Delta\alpha$, $\tan\Delta\alpha \approx \Delta\alpha$, we have

$$\Delta\alpha \approx \tan\Delta\alpha = \frac{(x_1 - x_2)\Delta y_2 - y_2(\Delta x_1 - \Delta x_2)}{(x_1 - x_2)^2 + (x_1 - x_2)(\Delta x_1 - \Delta x_2) + y_2^2 + y_2\Delta y_2} \quad (1)$$

We want to compute the expectation of $|\Delta\alpha|$. Suppose Δx_1 , Δx_2 , and Δy_2 are independent, and $-4 \leq \Delta x_i, \Delta y_2 \leq 4$, $i = 1, 2$, and Δx_i and Δy_2 are all integers, then

$$g(x_1, x_2, y_2) = E\{|\Delta\alpha|\} \approx \sum_{\Delta x_1=-4}^4 \sum_{\Delta x_2=-4}^4 \sum_{\Delta y_2=-4}^4 (|\tan\Delta\alpha| \times p(\Delta x_1)p(\Delta x_2)p(\Delta y_2)) \quad (2)$$

Suppose $p(\Delta x_1)$, $p(\Delta x_2)$ and $p(\Delta y_2)$ are discrete uniform distributions in $[-4, +4]$. Let $0 < x_1, y_2, |x_2| < L$, where L is the maximum value of these variables in the image (in our experiments, $L = 150$). We compute $g(x_1, x_2, y_2)$ at each point (x_1, x_2, y_2) based on whether α is the minimum, median or maximum angle in the triangle.

From Table 1, we observe: (a) the percentages of the expectation of changes of angles less than the threshold for minimum angle and median angle are always greater than that for the maximum angle; (b) $2''$ is a good threshold for dealing with changes caused by minutiae locations uncertainty in $[-4, +4]$. The percentages of the expectation of changes of angles less than $2''$ are 93.2% and 87.3% for α_{\min} and α_{med} , respectively. Using other distributions for $p(\Delta x_1)$, $p(\Delta x_2)$ and $p(\Delta y_2)$, we find the results similar to that in Table 1.

Table 1. Percentage of the expectation of changes of angles less than the threshold.

Angle s Type	Angle Change Threshold					
	1°	2°	3°	4°	5°	6°
α_{\min}	51.6	93.2	98.5	99.6	99.9	100.0
α_{med}	56.6	87.3	94.5	97.3	98.7	99.4
α_{\max}	1.0	67.7	87.3	94.2	97.2	98.7

2.2 Index Elements and Geometric Constraints:

2.2.1 Indexing Function: Index elements are the features that are used to construct the indexing function $H(\alpha_{\min}, \alpha_{\text{med}}, \phi, \gamma, \eta, \lambda)$. Note that only the hypotheses extracted within the tolerance used for the parameters in the indexing function H are passed on for checking the geometric constraints.

- **Angles α_{\min} and α_{med} :** Suppose α_i are three angles in the triangle, $i = 1, 2, 3$. Let $\alpha_{\max} = \max\{\alpha_i\}$, $\alpha_{\min} = \min\{\alpha_i\}$, $\alpha_{\text{med}} = 180^\circ - \alpha_{\max} - \alpha_{\min}$, then the label of the triplets in this triangle is such that if the minutia is the vertex of angle α_{\max} , we label this point as P_1 ; if the minutia is the vertex of angle α_{\min} , we label it as P_2 ; the last minutia is labeled as P_3 . We use α_{\min} and α_{med} as two elements in the indexing function H . $0^\circ < \alpha_{\min} \leq 60^\circ$ and $\alpha_{\min} \leq \alpha_{\text{med}} < 90^\circ$.

- **Triangle Orientation ϕ :** Let $Z_i = x_i + jy_i$ be the complex number ($j = \sqrt{-1}$) corresponding to the coordinates (x_i, y_i) of point P_i , $i = 1, 2, 3$. Define $Z_{21} = Z_2 - Z_1$, $Z_{32} = Z_3 - Z_2$, and $Z_{13} = Z_1 - Z_3$. Let $\phi = \text{sign}(Z_{21} \times Z_{32})$, where sign is the signum function and \times is the cross product of two complex numbers. $\phi = 1$ or -1 .

- **Triangle Type γ :** Let $\gamma = 4\gamma_1 + 2\gamma_2 + \gamma_3$, where γ_i is the feature type of point P_i , $i = 1, 2, 3$. If point P_i is an end point, then $\gamma_i = 1$, else $\gamma_i = 0$, $0 \leq \gamma \leq 7$.

- **Triangle Direction η :** Search the minutia from top to bottom and left to right in the image, if the minutia is the start point of a ridge or valley, then $v = 1$, else $v = 0$. Let $\eta = 4v_1 + 2v_2 + v_3$, v_i is the v value of point P_i , $i = 1, 2, 3$. $0 \leq \eta \leq 7$.

- **Maximum Side λ :** Let $\lambda = \max\{L_i\}$, where $L_1 = |Z_{21}|$, $L_2 = |Z_{32}|$, and $L_3 = |Z_{13}|$.

2.2.2 Geometric Constraints: These are used to eliminate any erroneous correspondences obtained from the above step. δ_1 , δ_2 and δ_3 tolerate rotation and errors in estimating the local orientation and δ_4 tolerates translation.

- Let points P_{21} , P_{32} , and P_{13} be the midpoint of line P_2P_1 , P_3P_2 and P_1P_3 , respectively, and point P_{123} be the centroid of the triangle $\Delta P_1P_2P_3$. Let $\phi_{21} = \psi_{21}$, $\phi_{32} = \psi_{32}$, $\phi_{13} = \psi_{13}$, and $\phi_{123} = \psi_{123}$, where ψ_{21} , ψ_{32} , ψ_{13} and ψ_{123} are the local orientation of points P_{21} , P_{32} , P_{13} and P_{123} , respectively. We have $|\phi - \phi'| < \delta_1$, where ϕ and ϕ' are ϕ_{21} , ϕ_{32} or ϕ_{13} in two impressions.

- Let ψ_i be the local orientation of point P_i , and $\omega_i = \psi_i - \psi_{123}$, we have $|\omega - \omega'| < \delta_2$, where $i = 1, 2, 3$, ω and ω' are ω_1 , ω_2 or ω_3 in two different impressions.

- Let $\theta_{21} = \text{angle}(Z_{21})$, $\theta_{32} = \text{angle}(Z_{32})$, and $\theta_{13} = \text{angle}(Z_{13})$, where $\text{angle}(Z)$ is the phase angle of the complex number Z . We have $|\theta - \theta'| < \delta_3$, where θ and θ' are θ_{21} , θ_{32} or θ_{13} of two different impressions.

- Let $Z_c = (Z_1 + Z_2 + Z_3) / 3$, we have $|Z - Z'| < \delta_4$, where Z and Z' are the Z_c in two different impressions.

The index score is computed according to the number of correspondences of triangles between the input image and images in the database.

2.3 Analysis of the Proposed Approach

2.3.1 Index Space and Discrimination: Since there is uncertainty associated with minutiae locations, a binning mechanism must be used. We use 0.5° as the bin size for angle α_{\min} and α_{med} , and 10 pixels for the λ . The bin size allows an appropriate tolerance for different distortions where α_{\min} and α_{med} tolerate shear and λ tolerates scale. According to the range of features, the number of entries for the index space H is $120 \times 180 \times 2 \times 8 \times 8 \times 20 = 55,296,000$ (assume $\lambda \leq 200$, there are 20 bins for λ). If there are 40 features in a image, then we have ${}^{40}C_3 = 40 \times 39 \times 38 / 6 = 9,880$ triangles in the image. However, if $\alpha_{\min} < \delta_\alpha$ or $\tau < \delta_\tau$, then the uncertainty of minutiae locations may have more effect on α_{\min} and α_{med} , so we do not use these triangles in the model-base, where τ is the minimum length of the sides in a triangle. Thus, only about 1/3 of the triangles are taken as models (for $\delta_\alpha = 10^\circ$, $\delta_\tau = 20$). Suppose these triangles are uniformly distributed in the index space, then this approach can host $55,296,000 / (9880/3) \approx 16790$ images with only one index in each entry. If there are N indices in each entry, then it can host $16790 \times N$ images.

2.3.2 Probability of False Indexing: Suppose (a) S is the size of the index space; (b) f_k is the number of triangles in the model-base for image I_k , and these triangles are uniformly distributed in the index space; (c) b is the search redundancy for each triangle in the test image; (d) v_k is the number of corresponding triangles between image I and I_k ; (e) f_t is the number of triangles for the test image; (f) p_0 is the probability to find a corresponding triangle in the index space for image I_k in a single search and p_1 is the probability in redundant search, then

$$p_0 = \frac{f_k}{S} \quad \text{and} \quad p_1 = 1 - (1 - p_0)^b \quad (3)$$

Hence, we can estimate $P\{v_k > T\}$ by the Poisson distribution with $\xi = f_t \times p_1$:

$$P\{v_k > T\} \approx 1 - e^{-\xi} \sum_{i=0}^T \frac{\xi^i}{i!} \quad (4)$$

When $T = 25$, $P\{v_k > T\} = 0.01$. So, $T = 25$ can be used as the threshold to reject a test image which has no corresponding image in the database. The triangles are not uniformly distributed in the model-base, however, we apply geometric constraints to eliminate erroneous correspondences, the value of T can be less than 25.

3 Experimental Results

3.1 Database: The data set used in our experiments has 1000 images. It includes 400 pairs of images and 200 single images. These images are collected under real-world conditions by a fingerprint optical sensor with the resolution of 300 DPI. The size of these images is 248×120 pixels. Each pair of images are different impressions of the same finger, one is used to construct the model-base, and the other is used to test the indexing performance. The range of the distortions between each pair of images are: translation (± 30 pixels), rotation ($\pm 30^\circ$), scale (1 ± 0.1), and shear (± 4

pixels). The single image data set is used to test the rejection performance. We subjectively classify the images according to their quality into three classes: good, fair and poor. The quality of images is determined visually based on the criteria, such as distortions between images, contrast, ridges continuity and width. A pair of images from each class is shown in Fig. 2. G_1 is the image in the database and G_2 is the test image. Notice the rotation between G_1 and G_2 in Fig. 2.1 and Fig. 2.2 and the dryness of the fingerprint reflected in the images in Fig. 2.3. Table 2 shows the composition of the database. Notice that most images in the database are of fair quality (33.2%) or poor quality (47.8%).

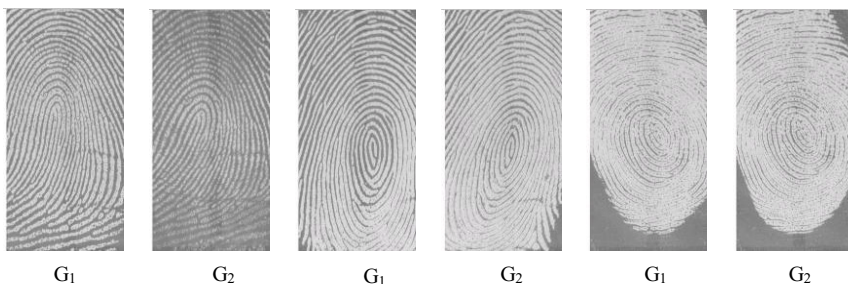


Fig. 2.1. Good images.

Fig. 2.2. Fair images.

Fig. 2.3. Poor images.

Table 2. Composition of the database.

	Images Quality			Summary
	Good	Fair	Poor	
# of Pairs of Images	78x2	124x2	198x2	400x2
# of Single Images	34	84	82	200
% in DB	19.0	33.2	47.8	100.0

3.2 Performance Evaluation Measures: A test image, which has a corresponding image in the database, is said to be correctly indexed if it has enough corresponding triangles in the model-base and the correct corresponding image appears in a shortlist of hypotheses obtained by the indexing approach. The Correct Index Power (CIP) is defined as the ratio of the number of correctly indexed images to the number of images used to construct the model-base. A test image, which has no corresponding image in the database, is said to be correctly rejected if it does not have enough corresponding triangles in the model-base. The Correct Reject Power (CRP) is defined as the ratio of the number of correctly rejected images to the number of the test images that do not have corresponding images in the database.

3.3 Experimental Results: The parameters of the algorithm are: $\delta_1 = \delta_2 = \delta_3 = 30^\circ$, $\delta_4 = 50$, $\alpha_1 = \alpha_2 = 2^\circ$, and $T = 20$. The bin size of λ is 10. Minutiae are automatically extracted using a technique in [5].

Table 3 shows that images of different quality have different results using the proposed approach. The CIP of a single hypothesis for good quality image is 96.2%. As the quality of images becomes worse, the CIP decreases to 85.5% for fair and 83.3% for poor images. The CIP of a single hypothesis for the entire database is 86.5%. The most important result of our experiments is that the CIP for the top two hypotheses is

100% for good images, and for fair images and poor images, the CIP for the top five hypotheses are 99.2% and 98%, respectively. For the entire database, the CIP for the top nine hypotheses is 100%. Further, all the 200 single images are rejected by our approach, thus, CRP is 100%. On a ULTRA2 workstation, average time for correctly indexing or correctly rejecting a test case is less than 1 second. It is much less than that of repeating a verification process (1 second [3]) for each image in the database.

Table 3. Correct Indexing Power of experimental results.

Top N Hy- potheses		Image Quality			Sum. of the entire DB
		Good	Fair	Poor	
N	1	96.2	85.5	83.3	86.5
	2	100	92.7	92.9	94.3
	3	100	95.2	95.5	96.3
	4	100	97.6	97.5	98.0
	5	100	99.2	98.0	98.8
	6	100	99.2	98.5	99.0
	7	100	99.2	100	99.8
	8	100	99.2	100	99.8
	9	100	100	100	100

4 Conclusions

Our experimental results show that the proposed indexing approach can greatly reduce the number of candidate hypotheses. The CIP of the top five and the top nine hypotheses are 98.8% and 100.0% for the entire database, respectively. This provides a reduction by a factor of more than 40 for the number of hypotheses that need to be considered for detailed matching. The CRP is 100.0%. Both CIP and CRP together characterize the discriminating power of our indexing approach. Our approach based on triplets of minutiae is promising for identifying fingerprints under translation, rotation, scale, shear, occlusion and clutter.

Acknowledgments: This work is supported in part by a grant from Sony, DiMI and I/O software. The contents and information do not necessarily reflect the positions or policies of the sponsors.

References

1. R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, Fingerprint classification by directional image partitioning, *IEEE Trans. on PAMI*, Vol. 21, No. 5, pp. 402-421, May 1999.

2. A.K. Jain, L. Hong, S. Pankanti, and R. Bolle, An identity-authentication system using fingerprints, *Proc. of the IEEE*, Vol. 85, No. 9, pp.1364-1388, September 1997.

3. N.K. Ratha, K. Karu, S. Chen, and A.K. Jain, A real-time matching system for large fingerprint databases, *IEEE Trans. on PAMI*, Vol. 18, No. 8, pp. 799-813, August 1996.

4. R.S. Germain, A. Califano, and S. Colville, Fingerprint matching using transformation parameter clustering, *IEEE Computational Science and Engineering*, Vol. 4, No. 4, pp. 42-49, October/November 1997.

5. B. Bhanu, M. Boshra, and X. Tan, Logical templates for feature extraction in fingerprint images, *Proc. Int. Conf. on Pattern Recognition*, Vol. 2, pp. 850-854, September, 2000.

Twin Test: On Discriminability of Fingerprints

Anil K. Jain¹, Salil Prabhakar¹, and Sharath Pankanti²

¹ Dept. of Comp. Sci. and Eng., Michigan State University, East Lansing, MI 48824

² IBM T. J. Watson Research Center, Yorktown Heights, NY 10598

Abstract. Automatic identification methods based on physical biometric characteristics such as fingerprint or iris can provide positive identification with a very high accuracy. However, the biometrics-based methods assume that the physical characteristics of an individual (as captured by a sensor) used for identification are distinctive. Identical twins have the closest genetics-based relationship and, therefore, the maximum similarity between fingerprints is expected to be found among identical twins. We show that a state-of-the-art automatic fingerprint identification system can successfully distinguish identical twins though with a slightly lower accuracy than nontwins.

1 Introduction

A number of identification systems based on different biometric characteristics have been developed. As the biometrics-based identification is becoming more pervasive, there is a growing interest [12] in determining the distinctiveness of biometrics characteristics in order to establish the performance limits of such systems. The distinguishing nature of physical characteristics of a person is due to both the inherent individual genetic diversity within the human population as well as the random processes affecting the development of the embryo [11]. Since two individuals can be arbitrarily close with respect to their genetic constitution (e.g., identical twins), a pessimistic evaluation of identity discrimination based on biometrics may need to rely solely on an assessment of diversity in the traits due to random process affecting human development. Such an assessment strategy would necessarily rely on biometric samples from individuals who are identical/similar in their genetic constitution.



Fig. 1. Photograph of identical twin sisters.

The extent of variation in a physical trait due to random development process differs from trait to trait. By definition, identical twins can not be distinguished based on DNA. Typically, most of the physical characteristics such as body type, voice, and face are very similar for identical twins and automatic identification based on face and hand geometry will fail to distinguish them. See Figure 1 for a photograph of an identical twin pair. It is, however, claimed that identical twins can be distinguished based on their fingerprints, retina, thermogram, or iris patterns. The focus of this study is to quantitatively determine the similarity of fingerprints in identical twins. We further attempt to assess the impact of this similarity on the performance of fingerprint-based verification systems. Since both, human iris and angiogenesis follow a development pattern similar to fingerprints, we believe the results of this study may be qualitatively applicable to other biometric identifiers such as iris, retina and thermogram as well.

2 Fingerprints

Fingerprints are the pattern of ridges on the tip of our fingers. They are one of the most mature biometric technologies and are considered legitimate proofs of evidence in courts of law all over the world. Fingerprints are fully formed at about seven months of fetus development and finger ridge configurations do not change throughout the life except due to accidents such as bruises and cuts on the finger tips. More recently, an increasing number of civilian and commercial applications (e.g., welfare disbursement, cellular phone access, laptop computer log-in) are either using or actively considering to use fingerprint-based identification because of the availability of inexpensive and compact solid state scanners as well as its superior and proven matching performance over other biometric technologies.

An important question in fingerprint matching is: which characteristics of the fingerprints are inherited? A number of studies have shown a significant correlation in the fingerprint class (i.e., whorl, right loop, left loop, arch, tented arch) of identical twin fingers; correlation based on other generic attributes of the fingerprint such as ridge count, ridge width, ridge separation, and ridge depth has also been found to be significant in identical twins. In dermatoglyphics studies, the maximum generic difference between fingerprints has been found among individuals of different races. Unrelated persons of the same race have very little generic similarity in their fingerprints, parent and child have some generic similarity as they share half the genes, siblings have more similarity and the maximum generic similarity is observed in the monozygotic (identical) twins, which is the closest genetic relationship [5].

Monozygotic twins are a consequence of division of a single fertilized egg into two embryos. Thus, they have exactly identical DNA except for the generally undetectable micromutations that begin as soon as the cell starts dividing. Fingerprints of identical twins start their development from the same DNA, so they show considerable generic similarity. However, identical twins are situated in different parts of the womb during development, so each fetus encounters slightly different intrauterine forces from their siblings. As a result, fingerprints

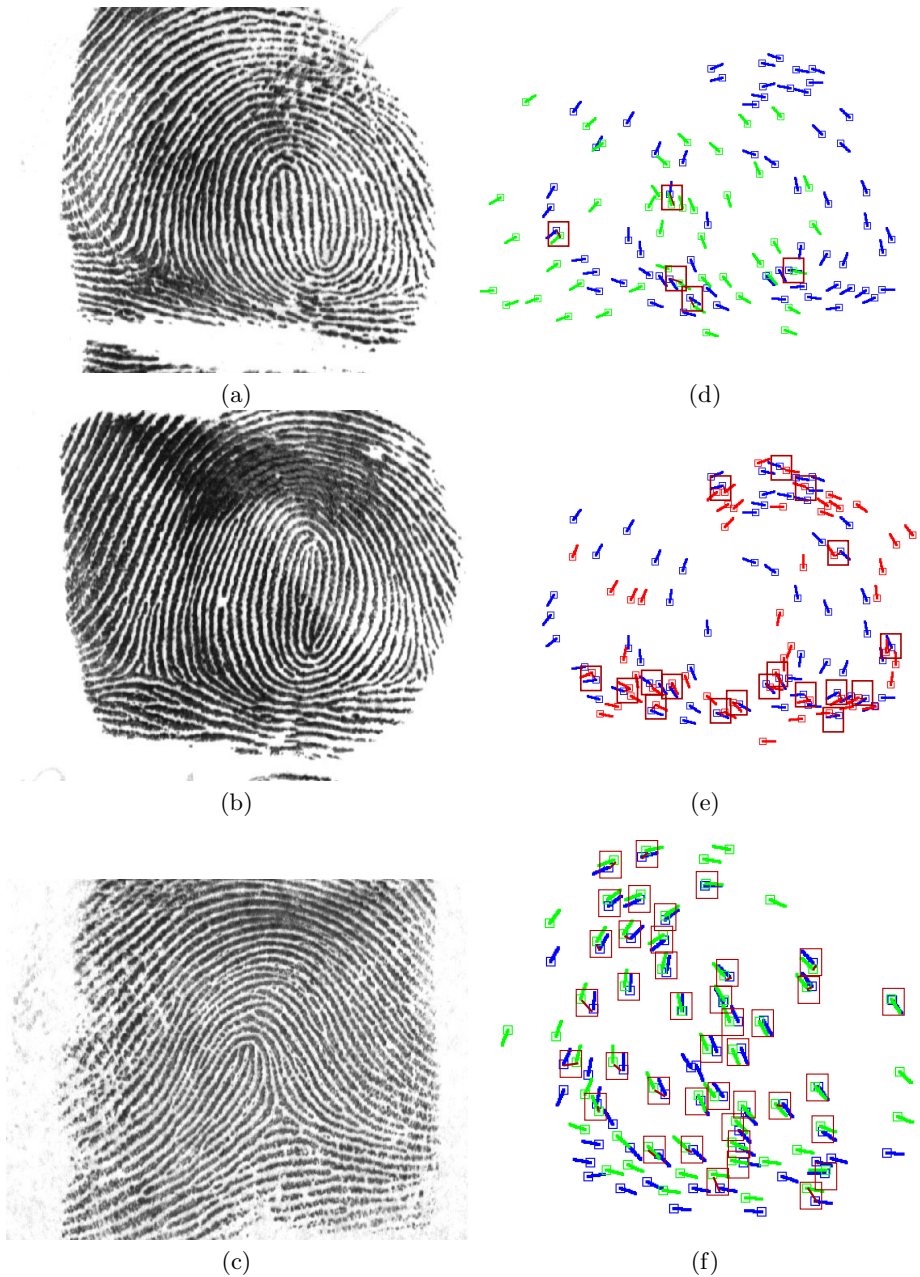


Fig. 2. Minutiae matching for twins. (a) and (b) are fingerprint images of identical twins while the fingerprint in (c) is from another person. (d) Minutiae matching for twin-nontwin (matching of (b) and (c), matching score = 3). (e) Minutiae matching for twin-twin (matching of (a) and (b), matching score = 38). (f) Minutiae matching for two impressions of the same finger (these images are not shown here) of a person (matching score = 487). The “matched” minutiae pairs are shown by bounding boxes.

of identical twins have different microdetails which can be used for identification purposes [4]. It is claimed that a trained expert can usually differentiate between the fingerprints of identical twins based on the minutiae (dis)similarity [6]. Thus, there is anecdotal evidence that minutiae configurations are different in identical twins but to our knowledge, no one has systematically investigated or quantified how minutiae information in identical twins is (un)related in the context of an automatic fingerprint-based authentication system. This paper focuses on analyzing the similarity between fingerprint minutiae patterns of identical twin fingers.

3 Experimental Results

An arbitrary subset of the rolled identical twin fingerprints collected for the NHLBI twin study [7] is used in our experiments. The fingerprints of the index fingers of 100 pairs of identical twins were scanned at 500 *dpi* resolution. Due to differences in paper quality and degradation of the print over time, several of these fingerprints are of poor quality. As a result, we used only 94 pairs of identical twin fingerprints in our study. Using the algorithm described in [8], we matched every fingerprint in our twin database with every other fingerprint. The twin-twin imposter distribution was estimated using 188 (94×2) matchings between the 94 twin fingerprint pairs whereas the twin-nontwin imposter distribution was estimated using 17,484 ($94 \times 93 \times 2$) matchings. The twin-twin imposter distribution was found to be slightly shifted to the right of the twin-nontwin distribution indicating that twin-twin fingerprints are generally more similar than twin-nontwin fingerprints. The two-sample Kolmogorov-Smirnov test [9] showed that the twin-twin and twin-nontwin distributions are significantly different at 99.9% level. A genuine distribution of matching scores is generally estimated by matching multiple fingerprint images of the same finger. Since we had access to only a single impression of the fingers in our twin database, we approximated the genuine distribution for twin-twin matching from the genuine distribution of the NIST9 CD No. 1 [10] fingerprint database which consists of 1,800 fingerprint images taken from 900 independent fingers, two impressions per finger. The ROC curve (Figure 3(a)) shows that, due to the similarity of twin fingerprints, the ability of the system to distinguish identical twins is lower than its ability to distinguish twin-nontwin pairs. An illustration is shown in Figure 4. However, contrary to claims made in popular press [11], the automatic fingerprint identification system can still be used to distinguish between identical twins without a drastic degradation in identification performance.

If an application demands an FAR of 1%, to safeguard against twin fraud, we can set the operating point of the system pessimistically at a threshold that corresponds to an FAR of $\sim 0.3\%$ for twin-nontwin matchings and an FAR of $\sim 1.0\%$ for twin-twin matchings. This threshold corresponds to an FRR of 3.5%. This means that in the worst case scenario (when all the people accessing the system are twins), the system will falsely accept 10,000 people out of one million at the expense of falsely rejecting 35,000 people. In the best case (when there

are no twins accessing the system), only 3·000 people will be falsely accepted while falsely rejecting 35·000 people. In practice, the system will falsely accept between 3·000 and 10·000 people (between 0·3% and 1%), depending upon the fraction of twins in our sample population of one million while falsely rejecting 35·000 people.

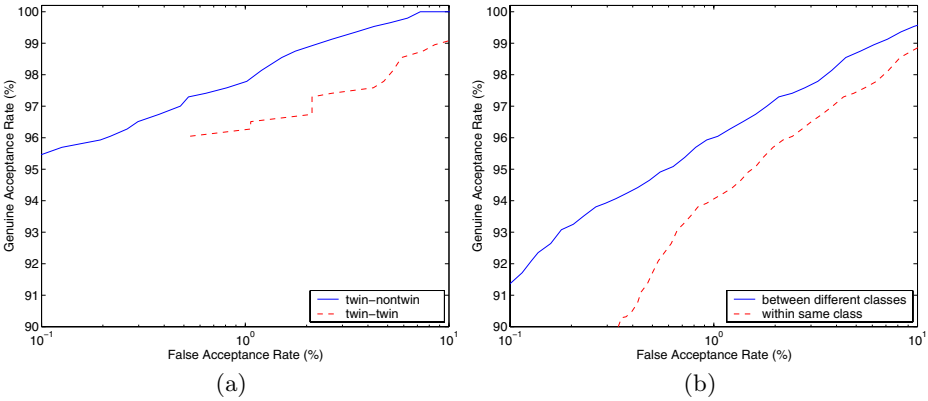


Fig. 3. ROC curves for (a) twin-twin and twin-nontwin minutiae matchings, and (b) effect of fingerprint class type on the matching score.

Dermatoglyphics studies have suggested that there is high class similarity in the fingerprints of identical twins. To confirm this claim, we manually classified the 94 pairs of identical twin fingerprints in our database into five classes (right loop, left loop, whorl, arch, and tented arch). The class correlation between the index fingers of identical twins is found to be 0·775 (fraction of identical twin pairs whose index fingerprints have the same class label). The natural proportion of occurrence of each of the five major classes of fingerprints in the index finger is 3252, 3648, 1703, 8616, and 8779 for whorl, right loop, left loop, arch, and tented arch, respectively. If we randomly choose two index fingerprint images from a large database, the probability that these two fingerprints will have the same class label is equal to 0·2718. Thus, there is only 0·2718 chance that two randomly chosen index fingers will have the same type which is much lower than the 0·775 chance that the fingerprints of two identical twins will have the same class label.

We believe that the global similarity of fingerprints (shown as class similarity) is, to a certain extent, responsible for the local similarity (shown in the matching performance). Consider two fingerprints that belong to the same class (e.g., right loop). Since the minutiae can exist only along the ridges (although at random locations), the matching score between these two fingerprints is likely to be higher than the matching score between two sets of random point patterns. To study the correlation of class information with the matching performance, we computed the genuine distribution from 3·600 matchings between the two

impressions of the same finger from 1,800 good quality fingerprint pairs from the NIST4 database [11]. The between-class and within-class distributions were computed from about 130,000 matchings each. The ROCs for between-class and within-class matchings are shown in Figure 3(b). Note that the magnitude of the shift between the two ROCs in Figure 3(b) is of the same order of magnitude as the one manifested in Figure 3(a). Hence, we conjecture that the larger similarity observed in identical twins is due to the high class correlation in their fingerprint types.

4 Conclusions

One out of every eighty births results in twins and one third of all the twins are monozygotic (identical) twins. Some identical twins have been reported to be involved in fraud, which can be called as “twin fraud”, since people mistake the identities of the identical twins. There have been cases reported where an identical twin was sentenced for a crime that was committed by his/her sibling [1]. Fertility treatments have resulted in an increase in the identical twin birth rate (in fact, according to a study by Robert Derom [12], the identical twin birth rate is about twice as high for women who use fertility drugs). Further, because of the medical advances in the treatment of premature babies, population of identical twins is increasing. We have shown that even though identical twin fingerprints have large class correlation, they can still be distinguished using a minutiae-based automatic fingerprint identification system; though with slightly lower accuracy than nontwins. Our results suggest that the marginal degradation in performance may be related to the dependence of the minutiae distribution on fingerprint class.

What are the implications of our empirical results in person identification applications? In authentication applications, marginal degradation in accuracy performance will have almost no effect on “evil” twins posing as impostors. In large scale fingerprint based identification applications, a small degradation in authentication accuracy may imply a significant degradation in the recognition accuracy. Further, if the degradation in the performance is dependent on the class correlation which in turn depends on the genetic constitution (as suggested by the dermatoglyphics studies), it may imply that benefits reaped by composition of ten-finger information may have been overestimated in the literature. Further, the magnitude of performance degradation of a minutiae-based fingerprint matcher may depend upon the genetic relationship among a target population corpus. Both of these effects may need further investigation; more research is necessary for class-independent minutiae-based matchers. Finally, the case of fingerprint classification for the binning of population to increase efficiency of fingerprint based search may not be very effective in genetically related population.

References

1. D. Costello, "Families: The Perfect Deception: Identical Twins", *Wall Street Journal*, February 12, 1999.
2. Problem Idents. <http://onin.com/fp/problemidents.html>.
3. R. G. Steen, *DNA and Destiny: Nature and Nurture in Human Behavior*, New York: Plenum Press, 1996.
4. N. L. Segal, *Entwined Lives: Twins and What They Tell Us About Human Behavior*, Plume, New York, 2000.
5. H. Cummins and Charles Midlo, *Fingerprints, Palms and Soles: An Introduction to Dermatoglyphics*, Dover Publications, Inc., New York, 1961.
6. E. P. Richards, "Phenotype vs. Genotype: Why Identical Twins Have Different Fingerprints?", http://www.forensic-evidence.com/site/ID_Twins.html.
7. E. Splitz, R. Mountier, T. Reed, M. C. Busnel, C. Marchaland, P. L. Roubertoux, and M. Carlier, "Comparative Diagnoses of Twin Zygosity by SSLP Variant Analysis, Questionnaire, and Dermatoglyphics Analysis," *Behavior Genetics*, pp. 56-63, Vol. 26., No. 1, 1996.
8. A. K. Jain, L. Hong, S. Pankanti, and R. Bolle, "An Identity Authentication System Using Fingerprints," *Proc. IEEE*, Vol. 85, No. 9, pp. 1365-1388, 1997.
9. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C, The Art of Scientific Computing*, Cambridge University Press, 1992.
10. NIST Special Databases 4 and 9. <http://www.nist.gov/srd/special.htm>

An Improved Image Enhancement Scheme for Fingerprint Minutiae Extraction in Biometric Identification

D. Simon-Zorita, J. Ortega-Garcia, S. Cruz-Llanas,
J.L. Sanchez-Bote, and J. Glez-Rodriguez

Biometric Research Lab., ATVS-EUIT Telecomunicacion
Universidad Politecnica de Madrid, Spain

dsimon@diac.upm.es

<http://www.atvs.diac.upm.es>

Abstract. In this paper a complete algorithmic scheme for automatic fingerprint recognition is presented. In [3] an identification/verification system is described. The whole recognition process is accomplished in two stages: in the first one, biometric characteristics of fingerprints are extracted (characteristics denoted as *minutiae*, which represent basically the beginning, end or bifurcation of a ridge), and in the second stage, those fingerprints will be matched with templates belonging to the test database. In this paper, some improving alternatives regarding the first stage, namely the image enhancement process, are proposed, consequently leading to an increase of the reliability in the minutiae extraction stage. Conclusions in terms of *Goodness Index* (GI) are provided in order to test the global performance of this system.

1 Introduction

One of the most important tasks considering an automatic fingerprint recognition system is the minutiae biometric pattern extraction from the captured image of the fingerprint. In some cases, the fingerprint image comes from an inked fingerprint; in other cases, the image is obtained directly scanning the fingerprint. Due to imperfections of the acquired image, in some cases certain minutiae can be missed by the extraction algorithm, and in other cases spurious minutiae can be inserted [3,4,7]. Image imperfections can also generate errors in determining the coordinates of each true minutia and its relative orientation in the image. All these facts make remarkable decrease of the recognition system reliability, since fingerprint recognition is based on the comparison, within some tolerance limits, between the testing biometric pattern and the stored pattern. The algorithms developed and proposed in this paper for the image enhancement process have been tested with both inked and scanned fingerprints, leading to an evaluation of results in terms of *Goodness Index* (GI) [5,6], and allowing therefore a direct comparison between the automatically-extracted minutiae pattern and the pattern obtained by expert peer-inspection. Some conclusions regarding both image enhancement algorithms and GI results are also presented.

2 Fingerprint Image Enhancement

To apply the proposed automatic minutiae extraction algorithmic solution, we have used fingerprints from the database DB 4 NIST Fingerprint Image Groups. The NIST images derive from digitized inked fingerprints, each one consisting of 512x512 pixels, in 8-bit gray scale. The medium-quality of these inked fingerprints makes imperfections arise: non-uniformity of the ink density, appearance of non-printed areas, and also the existence of stains. The system has also been evaluated with scanned fingerprints, in which image brightness and contrast controls provide an increase of image quality, improving the computational efficiency of the global enhancement algorithm. The 100SC Precise Biometric fingerprint scanner has been used to acquire a small-sized fingerprint database (ATVS database), in order to compare the results with those obtained in the evaluation of the NIST fingerprints. ATVS database consists of 100 users with 300x300 pixel images, in 8-bit gray scale. Next, in order to accurately extract the minutiae of a fingerprint, the improved sequence of stages in which the complete outlined process consists is described.

2.1 Image Normalization

The objective of this stage is to decrease the dynamic range of the gray scale between ridges and valleys of the image in order to facilitate the processing of the following stages. The normalization factor is calculated according to the mean and the variance of the image [5]. Figure 1 shows the original NIST f05 image and the normalized fingerprint.

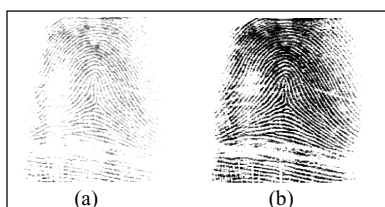


Fig. 1. (a) NIST f05 original image.
(b) Normalized fingerprint.

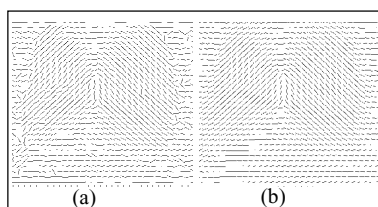


Fig. 2. (a) Fingerprint oriented and
(b) re-aligned fields.

2.2 Calculation of the Orientation Field

The orientation field represents the local orientation of the ridges contained in the fingerprint. In order to estimate it, the image is divided in 16x16 pixel blocks and the gradient is calculated every pixel, in x and y coordinates. Due to the inherent computational load of the recognition process, it is enough to apply a mask of 3x3 pixels for the gradient calculation at each pixel. From the gradient information the

orientation angle is estimated through the minimum square adjustment algorithm given in [1-3,5]. Often, in some blocks, the orientation angle is not correctly determined, due to background noise and ridges and valleys damages, caused by impression lacks of certain image areas. Therefore, as significant local angle variations between adjacent blocks cannot exist, a new spatial low-pass filtering is applied to the estimated oriented field to correctly re-align all the segments. The filter mask size used is 5×5 pixels, with fixed coefficient weighting of $1/25$. Figure 2(a) shows the resulting orientation field obtained from gradient calculation. Figure 2(b) shows the re-aligned field obtained after the spatial low pass filtering. This orientation field will fix the adaptive filters parameters in successive stages.

2.3 Selection of the Interest Region

Since the image has background noise, the algorithm may generate minutiae outside the fingerprint area. To avoid this problem, the image area defined by all the 16×16 blocks, in which a high variance of the gray level in the normal direction of the ridges exists, is selected. Thus, the normal orientation field of the ridges is previously estimated. After that, the noisy area of the image, to be excluded in the following steps, is defined by low variance in all directions [6]. In figure 3 it is shown the variance for the NIST f05 fingerprint and the interest region derived for subsequent stages.

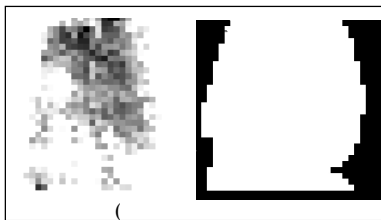


Fig. 3. (a) Fingerprint variance and (b) interest region.

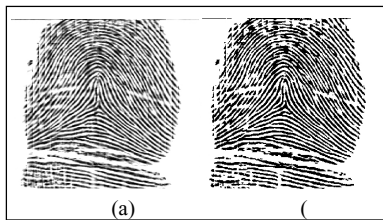


Fig. 4. (a) Filtered image with a spatial mask. (b) Obtained binary image.

2.4 Ridge Extraction

In order to decide whether a single pixel belongs or not to a given ridge, it is necessary to filter the fingerprint image with two adaptive masks, both capable to increase the gray level in the normal direction of the ridge [1-3]. The orientation of the mask is adapted within each 16×16 block, depending on the angles obtained from the orientation field of figure 2(b). If the gray level of a pixel exceeds a threshold in the two filtered images, it is thus considered that the pixel belongs to a ridge; otherwise, it is assigned to a valley, producing a binary image of the fingerprint. The dimensions of the mask are $L \times L$ (typ., $L=5$), and they are defined by the functions given in (1,2).

$$h_1(u,v) = \begin{cases} \frac{1}{\sqrt{2\pi}\delta} e^{-\left(\frac{u-u_0}{\delta}\right)^2}, & \text{if : } u_0 = E[(v_c - v) \text{ctg}(\theta) + u_c] \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$h_2(u,v) = \begin{cases} \frac{1}{\sqrt{2\pi}\delta} e^{-\left(\frac{v-v_0}{\delta}\right)^2}, & \text{if : } v_0 = E[(u_c - u) \text{tg}(\theta) + v_c] \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$\forall u, v \in [1, L]$, where u and v are the coordinates of a pixel in the mask; (u_c, v_c) , is the center of the mask; θ , is the orientation angle of the ridge in each image block, and δ , is a parameter to adjust the function mask to the width of the ridge. Figure 4(a) shows the filtered image with one of the spatial masks. Figure 4(b) represents the binary image obtained after a threshold is applied, producing smoother ridge borders. For good quality fingerprint images, as in scanned fingerprints, the previous filtering process is simplified using just a single mask, with an spatially-oriented impulse (unit amplitude and zero width function), also spatially adapted to the angle of the ridges, and then also binaryzed through a threshold.

2.5 Ridge Profiling

To simplify the processing of the following steps, a new image filtering to profile the fingerprint ridges and eliminate the stains of certain areas is applied. In order to accomplish this process, the low frequency components are first extracted and then subtracted to the original image, providing the high frequency components necessary to profile the ridges, as can be derived from (3):

$$p[u,v] = f[u,v] + \lambda \cdot f_H[u,v] = f[u,v] + \lambda \cdot (f[u,v] - f_L[u,v]) \quad (3)$$

where $p[u,v]$, is the resulting profiling image; $f[u,v]$, is the binary image; $f_H[u,v]$ and $f_L[u,v]$ are, respectively, the high and low frequency images; and λ is a factor ($\lambda > 0$), that determines the degree of profiling. In figure 5(a) the resulting filtered image is shown. An additional filtering can be applied to eliminate the spurious ridges due to stains in the image. Thus, a unit impulse mask is used, capable to locally adapt its orientation to the ridge orientation. The resulting binary image is shown in figure 5(b).

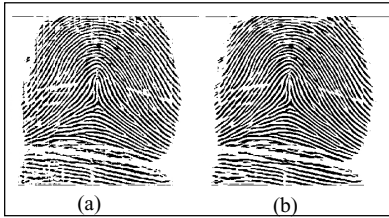


Fig. 5. (a) Image after first profiling filtering. (b) Image after filtering with spatial mask.

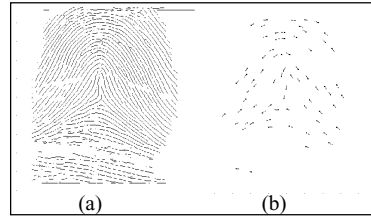


Fig. 6. (a) Image after thinning and imperfection removal. (b) Minutiae after cluster elimination.

2.6 Thinning

In this step two consecutive fast parallel thinning algorithms are applied, in order to reduce to a single pixel the width of the ridges in the binary image. These operations are necessary to simplify the subsequent structural analysis of the image for the extraction of the fingerprint minutiae. The thinning must be performed without modifying the original ridge structure of the image. During this process, the algorithms cannot miscalculate beginnings, endings and/or bifurcation of the ridges, neither ridges can be broken.

2.7 Imperfection Removal

After thinning, depending on the image quality, the structural imperfections of the original fingerprint remain in certain degree. This results in breaking ridges, spurious ridges and holes; therefore, it is necessary to apply an algorithm for removing all the lines not corresponding to ridges and an algorithm to connect all the broken ridges. Figure 6(a) shows the thinned image obtained once the algorithms for thinning and imperfection removal are applied [4].

2.8 Minutiae Extraction

In the last stage, the minutiae from the thinning image are extracted, obtaining accordingly the fingerprint biometric pattern. This process involves the determination of: *i*) whether a pixel, belongs to a ridge or not and, *ii*) if so, whether it is a bifurcation, a beginning or an ending point, obtaining thus a group of candidate minutiae. Next, all points at the border of the interest region are removed. Then, since the minutiae density per unit area cannot exceed a certain value, all the candidate-point clusters whose density exceed this value are substituted by a single minutia located at the center of the cluster. Figure 6(b) shows the resulting minutiae pattern. Once the minutiae extraction process is concluded, the resulting biometric pattern contains, typ., 70~80 points.

3 Experimental Results

As it is described in [5,6], matching of the minutiae pattern obtained by visual inspection with the minutiae pattern obtained by the automatic extraction algorithm is accomplished. In order to evaluate this process, the *Goodness Index* (GI) of the extracted minutiae is defined in (4) where an 8x8 pixel tolerance box to evaluate matching between the minutiae in the two patterns is considered:

$$GI = \frac{\sum_{i=1}^r q_i (p_i - d_i - i_i)}{\sum_{i=1}^r q_i t_i} \quad (4)$$

where r , is the total number of 16x16 image blocks; p_i , is the number of minutiae paired in the i th block; d_i , is the number of deleted (missing) minutiae by the algorithm in the i th block; i_i , is the number of spurious inserted minutiae generated by the algorithm in the i th block; t_i , is the true number of minutiae in the i th block; and q_i , is a factor which represents the image quality in the i th block (good=4, medium=2, poor=1). A high value of GI indicates a high reliability degree of the extraction algorithm. The maximum value, GI=1, is reached when all true minutiae are detected and no spurious minutiae are generated. Table 1 presents both the GI values obtained with 10 inked medium-quality fingerprint images from the NIST database (left side), and also with 10 scanned high-quality fingerprint images from ATVS database (right side), allowing consequently to set in this last case the image-quality weighting factor q_i directly to 1. Nevertheless, in the NIST case, in order to determine the number of true minutiae, the previously proposed binary image skeleton (see section 2.4) has been automatically used. After this enhancing process, the image-quality weighting factor q_i for NIST fingerprints has been also set to 1. Following this procedure, the total number of true minutiae vary within a range of 100 to 140 for NIST fingerprints, and 60 to 80 for ATVS ones, considering in both cases the entire interest region. Under the experimental conditions mentioned above, the resulting values, in the case of both inked and scanned fingerprints, outperform those presented in [5,6]. As it is shown in table 1, the quoted GI values vary within a range of 0.24 to 0.61 for inked fingerprints, and 0.33 to 0.76 for scanned ones.

4 Conclusions

The reliability of any automatic fingerprint recognition system strongly relies on the precision obtained in the minutiae extraction process. In this sense, the image enhancement algorithms play a decisive role. The complete image enhancement process proposed has been tested on a group of 10 inked medium-quality fingerprint images and 10 scanned ones, in order to evaluate its reliability in terms of image enhancement determined by the GI. Several relevant contributions have been proposed in this paper: *i*) the image enhancing process through the local oriented filtering of the image with two adaptive masks in order to extract the ridges of the fingerprint, *ii*) the reliable extraction of the ridge orientation angle through a previous re-estimate of the calculated orientation field, *iii*) the filtering process with a single adaptive mask of pulses in the case of scanned fingerprints and, *iv*) the ridge profile process, mainly in the case of inked fingerprints, since its skeleton is better defined, reducing the size of the possible stains of the image. A more precise minutiae localization process has been implemented, reducing consequently the generation of spurious inserted minutiae. The consistent improvements found in the global image enhancement process lead to relatively high GI values, resulting in a competitive scheme compared to those previously proposed.

Table 1. GI values for NIST and ATVS fingerprints. *P* stands for paired , *D* for deleted , *I* for inserted , and *T* for true minutiae.

VIS1	P	D	I	T	GI	ATVS	P	D	I	T	GI
f09	119	25	6	144	0.61	01	25	0	6	25	0.76
s04	100	11	22	111	0.60	02	45	5	5	50	0.70
s20	81	19	3	100	0.59	03	45	1	15	46	0.65
f05	91	23	4	114	0.56	04	48	8	7	56	0.59
s10	153	39	8	192	0.55	05	44	5	11	49	0.57
f18	77	29	0	103	0.45	06	45	8	8	53	0.55
s23	74	24	10	98	0.41	07	50	10	8	60	0.53
s16	105	40	7	145	0.40	08	46	5	18	51	0.45
f02	111	49	6	160	0.35	09	38	10	9	48	0.40
f12	117	46	31	163	0.24	10	29	1	18	30	0.33

References

- [1] J. Bigun and G.H. Granlund, Optimal Orientation Detection of Linear Symmetry , First International Conference on Computer Vision, London, June 1987, IEEE Computer Society Press, Washington DC, pp. 433-438.
- [2] J. Bigun, G.H. Granlund, and J. Wiklund, Multidimensional Orientation Estimation with Applications to Texture Analysis and Optical Flow , IEEE-PAMI, Vol. 13, No. 8, pp. 775-790, Aug. 1991.
- [3] A. Jain, L. Hong, and R. Bolle, "On-Line Fingerprint Verification", IEEE-PAMI, Vol.19, No.4, pp. 302-314, Apr. 1997.
- [4] D.C. Douglas Hung, Enhancement and Feature Purification of Fingerprint Images , Pattern Recognition, Vol. 26, No. 11, pp. 1661-1671, Nov. 1993.
- [5] L. Hong, Y. Wan, and A. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation", IEEE-PAMI, Vol. 20, No. 8, pp. 777-789, Aug. 1998.
- [6] N.K. Ratha, S. Chen, and A. Jain, "Adaptive Flow Orientation-Based Feature Extraction in Fingerprint Images", Pattern Recognition, Vol. 28, No. 11, pp. 1657-1672, Nov. 1995.
- [7] N. Ratha, K. Karu, S. Chen, and A. Jain, "A Real Time Matching System for Large Fingerprint Databases", IEEE-PAMI, Vol. 18, No. 8, pp. 799-813, Aug. 1996.

An Analysis of Minutiae Matching Strength

Nalini K. Ratha, Jonathan H. Connell, and Ruud M. Bolle

IBM Thomas J. Watson Research Center
30 Saw Mill River Road, Hawthorne, NY 10532
{ratha,jconnell,bolle}@us.ibm.com

Abstract. In recent years there has been exponential growth in the use of biometrics for user authentication applications because biometrics-based authentication offers several advantages over knowledge and possession-based methods such as password/PIN-based systems. However, it is important that biometrics-based authentication systems be designed to withstand different sources of attacks on the system when employed in security-critical applications. This is even more important for unattended remote applications such as e-commerce. In this paper we outline the potential security holes in a biometrics-based authentication scheme, quantify the numerical strength of one method of fingerprint matching, then discuss how to combat some of the remaining weaknesses.

1 Introduction

Reliable user authentication is becoming an increasingly important task in the web-enabled world. The consequences of an insecure authentication method in a corporate or enterprise environment can be catastrophic, often leading to loss of confidential information, service denials, and issues with integrity of data and information contents. The value of a reliable user authentication is not limited to just computer access. Many other applications in everyday life also require user authentication, e.g. banking, immigration, and physical access control. These could also benefit from enhanced security. Automated biometrics technology in general, and fingerprints in particular, can provide a much more accurate and reliable user authentication method.

In this paper, we present in more detail the problems unique to biometric authentication systems and propose solutions to several of them. Though our analysis is very general and can be extended to other biometrics, we will focus on fingerprint recognition as an example throughout. In Section 2 we use a pattern recognition model of a generic biometrics system to help identify the possible attack points. In Section 3 we analyze the power of a minutia-based fingerprint system in terms of probability of a brute force attack being successful. In Section 4 we propose several techniques to ameliorate problems with “replay attacks”, the most likely source of attempted fraud.

2 Security of Biometrics

While automated biometrics can help to alleviate the problems associated with the existing methods of user authentication, hackers will still find the weak points

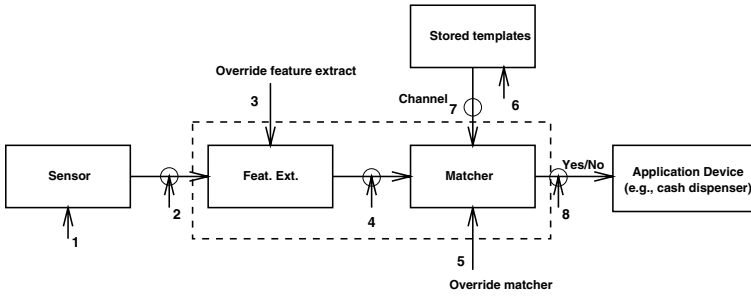


Fig. 1. Possible attack points in a generic biometrics-based system.

in the system and attack it at those points. Unlike password systems, which are prone to brute-force dictionary attacks, biometrics systems require substantially more effort to crack. Although standard encryption techniques are useful in many ways to prevent a breach of security, there are several new type of attacks are possible in the biometrics domain. If biometrics are used as a supervised authentication tool, this may not be a concern. But in remote unattended applications, such as web-based e-commerce applications, hackers may have enough time to make numerous attempts before being noticed, or may even be able to physically violate the remote client.

2.1 Pattern Recognition Based Threat Model

A generic biometric system can be cast in the framework of a pattern recognition system. The stages of such a generic system are shown in Figure 1. Excellent introductions to such automated biometrics can be found in [12]. Note that the password based authentication systems can also be put in this framework. The keyboard becomes the input device. The password encryptor becomes the feature extractor and the comparator becomes the matcher. The template database is equivalent to the encrypted password database.

There are in total eight basic sources of attack on such systems as described below. In addition, Schneier describes many other types of abuses of biometrics in [5]. In type 1 attack, a fake biometric (e.g., a fake finger, a face mask) is presented at the sensor. Resubmission of old digitally stored biometrics signal is the type 2 attack. In type 3 attack, the feature extractor could be attacked with a Trojan horse so that it would produce feature sets chosen by the hacker. After the features have been extracted from the input signal they are replaced with a different synthesized feature set (assuming the representation is known). If minutiae are transmitted to a remote matcher as in the case of using smartcards [4] to store the template than this threat is very real. In type 5 attack, the matcher is attacked to always directly produce an artificially high or low match score. The database of enrolled templates is available locally or remotely possibly distributed over several servers. The stored template attacker tries to modify one or more templates in the database which could result in authorization for a fraudulent individual, or at least denial of service for the person associated

with the corrupted template in type 6 attack. The templates from the stored database are sent to the matcher through a channel which could be attacked to change the contents of the templates before they reach the matcher in type 7 attack. Finally in type 8 attack, the authentication result can be overridden with the choice of result from the hacker.

2.2 Comparison to Passwords

A significant advantage of biometrics signals are that they are much longer in size than a password or pass phrase. They range from several hundred bytes to over a megabyte. Typically the information content of such signals is correspondingly higher as well. Simply extending the length of passwords to get an equivalent bit strength presents significant usability problems – it is nearly impossible to remember a 2K phrase and it would take an annoyingly long time to type in such a phrase anyhow (especially with no errors). Fortunately, automated biometrics can provide the security advantages of long passwords while still retaining the speed and simplicity of short passwords.

We also observe that the threats outlined in Figure 1 are quite similar in a password-based authentication system. For instance, all the channel attacks remain the same. However, in general fewer of them are typically covered. One such difference is that there is no “fake password” input detector equivalent to the fake biometrics described in threat 1 (although perhaps if the password was in some standard dictionary it could be deemed “fake”). Furthermore, in a password or token based authentication system no attempt is made thwart replay attacks (since there is no variation of the “signal” from one presentation to another). However, in an automated biometrics-based authentication system, one can go the extent of checking liveness of the input signal.

Another important difference concerns the matching subsystem. A password based method always provides a crisp result: if the passwords match, it grants access and otherwise refuses access. However the performance of a pattern recognition system in general is dependent on several factor such as the quality of input and enroll data along with the basic characteristics of the underlying algorithm. This is typically reflected in a graded overall match “score” between the submitted biometric and a stored reference. In a biometrics-based system, we can purposely set a threshold on the score to directly control the false accept and false reject rates. Inverting this, given a good matching score the system can guarantee that the probability of signal coming from a genuine person is significantly high. Such a calibrated confidence measure can be used to tackle non-repudiation support – something that passwords cannot provide.

3 Brute Force Attacks

In this section we show the relationship between the number of brute force attack attempts (point 4 in Figure 1) as a function of number of minutiae that are expected to match in the matcher subsystem. Generating all possible images (point 2) to guess the matching fingerprint image has a much larger search space and hence would be an even harder problem.

If reference print has N_r minutiae and each minutiae has d possible directions and one of K possible sites, then the probability that a randomly generated minutia will match one of the minutiae in the reference print in both site and direction is given by $p_{est} = \frac{N_r}{Kd}$. However, when generating random minutiae it is not desirable to generate two minutiae with the same site. So after $j-1$ minutiae have been generated, the probability that the j^{th} minutiae will match could be as high as $p \leq \frac{N_r}{(k-j+1)d}$. So to be conservative while generating N_q random minutiae we can assume each has matching probability $p = p_{hi} = \frac{N_r}{(K-N_q+1)d}$. The chance of getting exactly t of N_q generated minutiae to match is therefore given by $P_{thresh} = p^t(1-p)^{N_q-t}$. This break down for small K because the minutia matching probability changes depending on how many other minutiae have already been generated as well as on how many of those minutiae have matched. There are a number of ways of selecting which t out of the N_r minutiae in the reference print are the ones that match. Thus the total match probability becomes:

$$P_{exact} = \binom{N_r}{t} p^t (1-p)^{N_q-t} \quad (1)$$

But matches of m or more minutiae typically count as verification, so we get

$$P_{ver} = \sum_{t=m}^{N_q} \binom{N_r}{t} p^t (1-p)^{N_q-t} \quad (2)$$

For convenience, let us assume that $N_q = N_r = N$, so the above equation can be rewritten as

$$P_{ver} = \sum_{t=m}^{N_q} \binom{N}{t} p^t (1-p)^{N-t}, \quad (3)$$

since p is fairly small in our case, we can use a Poisson approximation to the binomial PDF.

$$P_{ver} = \sum_{t=m}^N \frac{(Np)^t e^{-Np}}{t!} \quad (4)$$

This summation is usually dominated by its first term. So neglecting all but the first term we find:

$$P_{ver} = \frac{(Np)^m e^{-Np}}{m!} \quad (5)$$

Because m is large, we can use Stirling's approximation and the equation can be written as

$$P_{ver} = \frac{(Np)^m e^{-Np}}{\sqrt{2\pi m} e^{-m} m^m} \quad (6)$$

For a value of $m=25$, we roughly have 82 bits of information content in this representation. This is equivalent to a nonsense password which is 16 characters long (like “m4yus78xpmks3bc9”).

We make several important observations. It can be seen in the simplistic and the complex model computations that if we have other local characteristics that can be attached to a minutia, then the probability of a brute force attack can be much lower through a brute force method (d is larger so p is smaller). And if the extent of spatial domain is increased (K is larger so p is smaller), the strength also increases. There is also a strong dependence on N , the overall number of minutiae in a fingerprint. For the best security, this number needs to be kept as low as possible – spurious minutiae from poor images are particularly detrimental.

4 Replay Attacks

As discussed earlier, one source of attack is the fraudulent resubmission of previously intercepted biometrics or biometric feature sets. Fortunately, several image-based techniques can be applied to address this problem. We describe two solutions based on our earlier work to thwart such replay attacks.

4.1 Image Based Challenge/Response Method

Standard cryptographic techniques, though mathematically strong, are computationally very intensive and require maintaining a secret key base for a large number of sensors. Moreover, encryption techniques cannot check for liveness of a signal. An encryptor will accept an old stored image as readily as a fresh one. Similarly a hash or digital signature of a signal does not check for its liveness, only its integrity.

Our proposed solution works as follows. At the user terminal or system, the transaction gets initiated. The transaction server then generates a pseudo-random challenge for the transaction and sends it to the intelligent sensor. Note that we assume that the transaction server is assumed to be secure. The sensor acquires a signal at this point of time and computes a response to the challenge based on the new biometric signal. For instance, a typical challenge might be “3, 10, 50”. The integrated processor might then select the 3rd, 10th and 50th pixel values from the new image to generate an output response such as “133, 92, 176”. This could be checked by the server to make sure that the client not only knew the correct response function, but also that the client was using the same image as received by the server.

By integrating the responder onto the same chip as the sensor it is just about impossible to inject a fake image (point 2 attack). Many silicon fingerprint scanners will be able to exploit the proposed method as they can integrate a processor without much effort. More details are available in [3].

4.2 WSQ-Based Data Hiding

Fingerprints are typically compressed with a wavelet technique called WSQ. Such compressed fingerprint images are then transmitted over a standard encrypted channel. Yet because of the open compression standard, transmitting a WSQ compressed image over the Internet is not particularly secure. If a compressed fingerprint image bit-stream can be intercepted (and decrypted), it can then be easily decompressed using readily available software. This potentially allows the signal to be saved and fraudulently re-used.

One way to enhance security is to use data-hiding techniques to embed additional information directly in compressed fingerprint images. For instance, assuming that the embedding algorithm remains inviolate, the service provider can look for an appropriate watermark to check that the submitted image was indeed generated by a trusted machine. Or the server might look for the response to a challenge as proposed in Section 3. The method proposed here (see [\[1\]](#) for more details) hides such messages with minimal impact on the decompressed appearance of the image. Moreover, the message is not hidden in a fixed location (which would make it more vulnerable to discovery) but is, instead, deposited in different places *based on the structure of the image itself*.

5 Conclusions

The weakest link in secure system design is user authentication. Biometrics can demonstrably improve this in terms of raw strength. And, for the same level of security, biometrics are preferable to passwords on the basis of user convenience alone. However, care must still be taken to prevent break-ins and special biometric-related issues must be understood. In particular, replay attacks must be guarded against. We proposed several methods, including an intelligent sensor challenge/response method and a data hiding technique for compressed signals, to bolster this aspect of such systems.

References

1. B. Miller, "Vital signs of Identity", IEEE Spectrum, February 1994, pp. 22–30.
2. A. Jain, L. Hong, and S. Pankanti, "Biometrics Identification", Communications of the ACM, February 2000, pp. 90–98.
3. B. Schneier, "The uses and abuses of biometrics". Communications of the ACM, August 1999, Vol. 42, No. 8, pp. 136.
4. N. K. Ratha and R. M. Bolle, "Smartcard based authentication", in Biometrics: Personal Identification in Networked Society (Eds. A. Jain, R. Bolle and S. Pankanti), Kluwer, 1999, pp. 369–384.
5. N. K. Ratha, J. H. Connell, and R. M. Bolle, "A biometrics-based secure authentication System", Proc. of the AutoID 99, Oct. 1999, pp. 70–73.
6. N. K. Ratha, J. H. Connell, and R. M. Bolle, "Secure data hiding in wavelet compressed fingerprint images", Proc. of the ACM Multimedia Workshop on Multimedia and Security, Nov. 2000, pp. 127–130.

Curvature-Based Singular Points Detection

Wai Mun Koo and Alex Kot

School of Electrical and Electronic Engineering
Nanyang Technological University
Singapore, 639798
P7248958F@ntu.edu.sg

Abstract. Singular Points Detection or more commonly known as 'Core Points Detection', is an important process in most fingerprint verification and identification algorithms for locating reference points for minutiae matching and classification. In this paper, we propose a new algorithm for singular points detection, which is based on scale-space analysis and curvature properties of the flow patterns of the fingerprint. The algorithm starts by examining the curvature of the fingerprint image at the coarsest scale and zoom in until the actual resolution of the image is reached. Experimental results show that the proposed algorithm is able to locate singular points in fingerprint with high accuracy.

1 Introduction

Fingerprint can be viewed as flow-like texture where the ridges and valleys combinations form the flow patterns of the image. A study of these flow patterns reveals many interesting features of the fingerprint (e.g. minutiae, ridge frequency, core and delta points etc). Core and delta points are unique landmarks for a fingerprint image though they are not always necessary to be present. They appear as unique points in a fingerprint image that can be used as reference points for matching and classification process [1-4]. In this paper, we introduce a new algorithm for the singular points detection, which is based on a scale-space analysis and the curvature properties of the flow patterns of the fingerprint. The algorithm starts by examining the curvature of the fingerprint image at the coarsest scale. It is followed by zooming in until the actual resolution of the image is reached. A very commonly used approach proposed by A.K. Jain [2] suggested a scheme that identifies singular points based on a Poincare Index computed for each block in the fingerprint image. Variations of the Poincare Index method are suggested in [1] and [5]. It is observed that these algorithms usually quantize the orientation fields into sectors to reduce the number of computations required. This reduces the fidelity of the orientation field, resulting in the increase of distance errors of the detected locations from the 'true' locations. Thus quantizing the orientation fields is not appropriate for applications that need high accuracy for locations of the singular points. The following sections will describe an algorithm that is able to detect singular points with high accuracy.

2 Singular Points Detection Algorithm

The proposed algorithm performs curvature analysis on different scales of the fingerprint image to obtain the core and delta points positions. The process starts at the coarsest scale and iterates through different scale levels until the actual scale of the image is reached. At each level, the orientation and curvature of the ridges are computed and the rough locations for the singular points are located by detecting the peak curvature values in the curvature map. The rough locations and their surrounding neighboring blocks will be selected for further analysis at the next iteration such that better precision locations will be obtained. At the finest level, the types of the detected singular points are determined to complete the process. By performing the curvature analysis at different scale levels, the algorithm is capable of rejecting local noise at the coarser levels while providing precision to the result as the process iterates toward the higher resolution levels.

Orientation Calculation

The first step of the algorithm is to calculate the Local Ridge Orientation (LRO). Orientation is a critical input property for the computation of the curvature map. The accuracy of the curvature map strongly depends on the accuracy of the orientation values calculated. The algorithm proposed by M. Kass and A. Witkin [6] is adopted for the orientation calculation. Below are the steps for orientation calculation:

1. Divide the image into blocks of size $w \times w$.
2. Compute the gradient vector $G(i,j)$ using the Sobel operators.
3. Compute the orientation of each block using the equation below,

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \left(\frac{\sum_{i=1}^w \sum_{j=1}^w 2G_x G_y}{\sum_{i=1}^w \sum_{j=1}^w (G_x^2 - G_y^2)} \right) + \frac{\pi}{2}, \quad (1)$$

where G_x and G_y are the horizontal and vertical derivatives of the fingerprint image computed using the Sobel operator, respectively.

4. Smooth the computed orientations with a 3×3 box filter.

Curvature Computation

Curvature can be defined as the rate of change of orientation over spatial variation. It is often used in image analysis processes for corner and junction detection and texture analysis. In the fingerprint image context, curvature can be used to measure how fast the ridge orientation varies in different portions of the image which is an important parameter for locating the singular points. Based on computation that has been computing the rate of change of the orientation field in a vectorial model, we compute the curvature $C(i)$ using,

$$C(i) = \frac{1}{2n} \sum_{k=1}^n [1 - \cos(\theta_k - \theta_i)], \quad 0 \leq C(i) \leq 1 \quad (2)$$

The curvature computation in (2) is derived from the derivative of the orientation field within a neighboring region of certain point. θ_i is orientation of pixel i , θ_k are the orientations of the neighboring elements in a 3×3 region with pixel i as the center, and $n = 8$ is the total number of neighboring elements.

Locating Peak Curvature Points

After computing the curvature map, the next step is to locate the peak curvature points in the map where the singular points reside. This can be achieved by selecting those blocks in the curvature map that satisfy the following two conditions:

1. $C(i) \geq C(k)$ where $k=1$ to 8, denotes the 8 neighboring elements of i .
2. $C(i) \geq \alpha$ where α is a predetermined threshold value.

where $\alpha = C_{mean} + C_{SD}$, C_{mean} and C_{SD} are the mean and the standard deviation of the curvature, respectively.

Determination of Singular Point Type

The last step of the singular point detection algorithm is to determine the type of the singular points detected. We propose below a method to determine the type of singular point by analyzing the structure shape formed by the orientation field around a local neighborhood. First, the four internal angles formed by the orientation field in a 2×2 region (see Figure 1) are computed using Equation (3) to (6) below.

$O(i,j)$ A	$O(i+1,j)$ B
$O(i,j-1)$ D	$O(i+1,j-1)$ C

Figure 1: Orientation internal angles

$$\Theta_{A,B} = (O(i+1, j) - O(i, j) + 2\pi) \bmod 2\pi, \quad (3)$$

$$\Theta_{B,C} = O(i+1, j-1) - O(i+1, j) + \pi, \quad (4)$$

$$\Theta_{C,D} = (O(i, j-1) - O(i+1, j-1) + 2\pi) \bmod 2\pi, \quad (5)$$

$$\Theta_{D,A} = O(i, j) - O(i, j-1) + \pi, \quad (6)$$

Next, an analysis is carried out on the four internal angles computed and the type of the singular point is determined through the conditions stated in Table 1. The block is then assigned to be core or delta points accordingly.

Type	Conditions
Delta	3 angles $< \beta$, and 1 angle $\geq \beta$
Core	3 angles $> \gamma$, and 1 angle $\leq \gamma$
where β is chosen to be 0.8π , and γ is chosen to be 0.2π	

Table 1: Conditions for type determination

The steps for the whole process of the singular points detection algorithm can be summarized as below:

- (1) Divide the raw fingerprint image into blocks of size $w_i \times w_i$.
- (2) Compute the Local Ridge Orientation (LRO) for each block.
- (3) Compute the Curvature for each block using LRO obtained from step (2).
- (4) Locate the blocks that contain the singular points by using peaks detection algorithm on the curvature information obtained from step (3).
- (5) For each block found in step (4), if $w_i \times w_i = 1 \times 1$ then go to step (7), else label the block and its 8 neighboring blocks as effective regions.
- (6) Divide the labeled effective regions into blocks of size $w_{i+1} \times w_{i+1}$ where $w_{i+1} \times w_{i+1} < w_i \times w_i$. Repeat steps (2) to (5).
- (7) Determine the types of the detected singular points.
- (8) End of process.

Steps (1) to (5) perform Local Ridge Orientation calculation and curvature analysis on the raw fingerprint image across several scale-spaces. Step (7) performs type identification for the detected singular points at the pixel level. The block size chosen for the coarsest level is 11×11 , which covers the average width of a ridge and a valley. The subsequent block sizes chosen are 7×7 , 3×3 and 1×1 (where 1×1 is the actual pixel size). The block sizes are chosen such that the block size for the next iteration is about half the block size of current iteration.

3 Experimental Results

The fingerprint database used in the experiment consists of 600 fingerprint images from 120 fingers with 5 fingerprint images for each finger. The size of the fingerprint images is 300×300 pixels with resolution of 300dpi. The prints in the database are classified into 485 normal, 50 dry and 65 wet prints by visual inspection. The fingerprint images are captured using a capacitive fingerprint and are quantized into of 256 gray levels of color. In the experiment, we compare the performance of the proposed singular points detection algorithm with the Poincare Index method. The performances of both algorithms are evaluated by investigating their accuracy in detecting the positions of the singular points. The positions of the singular points for all the prints in the database are first manually located by visual inspection and used as references for comparison. The mean distance between the detected singular point positions and the reference positions are computed for each algorithm and the results are tabulated in Table 2. The results show that the proposed curvature-based singular points detection algorithm has lower mean distance error (or higher accuracy) than the Poincare Index method.

Fingerprint Category	D_{mean} (Proposed curvature analysis method)	D_{mean} (Poincare Index method)
Normal 3.29		13.85
Dry	5.03	16.28
Wet 4.31		15.92

Table 2: Mean Distance Error

D_{mean} is the mean distance error between the reference positions and the positions detected by the algorithm. Figure 2 below shows some fingerprint images with singular points detected using the proposed curvature-based algorithm.

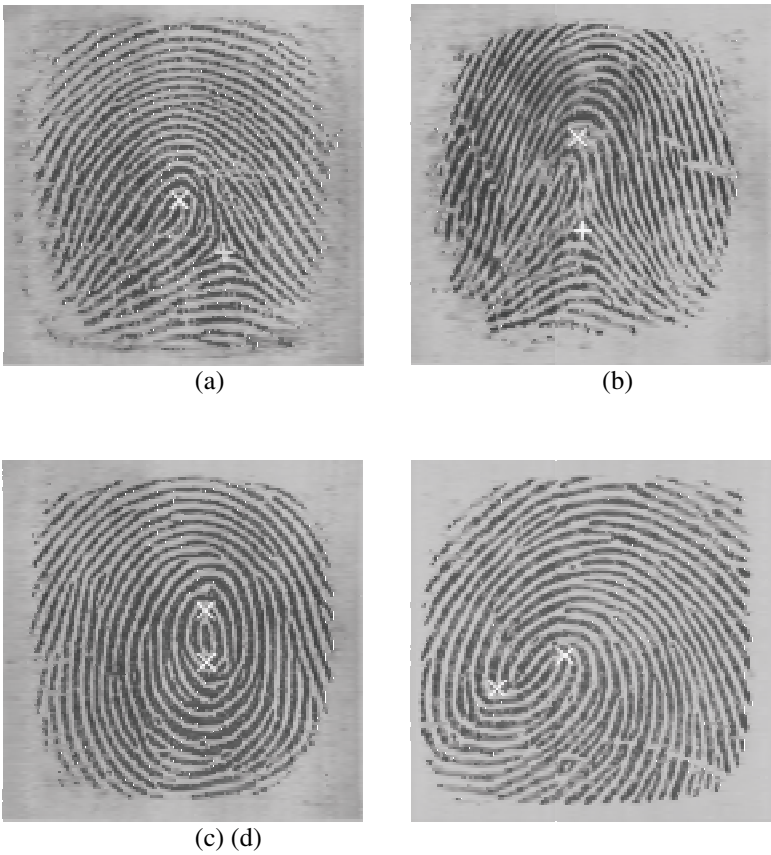


Figure 2: Results of Singular Points Detection

4 Conclusion

In this paper, we proposed a method to detect the singular points in the fingerprint by a curvature analysis method. The curvature of the ridge patterns in the fingerprint is analyzed in a multiresolution fashion so as to reject local noise at the coarser levels while providing precision to the result as the process iterates toward the higher resolution levels. Experimental results show that the proposed algorithm is able to detect singular points with higher accuracy.

Reference

1. B.H. Cho, J.S Kim, J.H Bae, I.G Bae and K.Y Yoo, Core-based Fingerprint Image Classification. ICPR'00, Spain, 2000.
2. L. Hong and A.K. Jain, Classification of Fingerprint Images, Proceedings of 11th Scandinavian Conference on Image Analysis, Jun 7-11, Kangerlussuaq, Greenland, 1999.
3. A. K. Jain, S. Prabhakar and L. Hong, A Multichannel Approach to Fingerprint Classification, *IEEE Transactions on PAMI*, Vol.21, No.4, pp. 348-359, Apr 1999.
4. Ballan M., Sakarya F.A., Evans B.L., A fingerprint classification technique using directional images. Conf. Rec. of the 31th Asilomar Conf. on Signals, Systems & Computers, 1997, 101-104 vol.1.
5. Ching T.H., Zhuang Y.L., Tan C.L., Kung C.M., An effective method to extract fingerprint singular point. Proc. The Fourth International Conference/ Exhibition on High Performance Computing in the Asia-Pacific Region, 2000, pp. 696 -699 vol.2.
6. M. Kass, A. Witkin, Analyzing Oriented Patterns. *Comp. Vision, Graphics and Image Proc.* vol. 37 pp. 362-385, 1987.

Algorithm for Detection and Elimination of False Minutiae in Fingerprint Images

Seonjoo Kim, Dongjae Lee, and Jaihie Kim

Department of Electrical and Electronics Engineering, Yonsei University, Seoul, Korea
sjkim23@seraph.yonsei.ac.kr

Abstract. A common problem in fingerprint recognition is the existence of false minutiae which increase both FAR and FRR in fingerprint matching. In this paper, a robust minutiae postprocessing algorithm is proposed. Unlike most algorithms which use simple distance and connectivity criteria for postprocessing, we also used orientation and flow of ridges as the key factor for postprocessing to avoid eliminating true minutiae while postprocessing. It is shown by the experiments that our postprocessing algorithm improves the minutiae extraction accuracy and the performance of the matching process.

1 Introduction

Most fingerprint recognition systems are based on minutiae matching [1]. Minutiae are local discontinuities of fingerprints and are restricted to two types : ridge ending and ridge bifurcation [2, 3]. A common problem in fingerprint recognition is the existence of false minutiae which increase both FAR and FRR in fingerprint matching. Therefore, the enhancement of the fingerprint image and the false minutiae elimination form an important part of the system. However, most of the researches emphasized on the fingerprint image enhancement and the false minutiae elimination process was based on simple distance and connectivity criteria [1, 2, 3, 4]. But the problem with such simple approaches is that it eliminates true minutiae while eliminating false minutiae. Xiao and Raafat proposed in [5], a minutiae postprocessing algorithm based on both statistical and structural information. However, their method relies heavily on connectivity which makes it complex and unreliable to bad quality fingerprints. Also, specific structural informations were not given.

In this paper, we present an efficient minutiae postprocessing algorithm. The goal of our minutiae extraction algorithm is to remove as many false minutiae as possible while retaining true minutiae. The goal is achieved by postprocessing minutiae based on not only the minutiae distance and connectivity but also using the orientation and flow of ridges as the key factor.

Rest of the paper is organized as follows. Section 2 briefly describes the adopted preprocessing procedures. Section 3 describes the proposed minutiae postprocessing procedures. The performance of the proposed algorithm is shown by experiments in Section 4. Finally, Section 5 contains conclusion.

2 Preprocessing and Minutiae Extraction

2.1 Preprocessing

Preprocessing procedures necessary for minutiae extraction are shown in Fig. 1.



Fig. 1. Preprocessing Procedures.

The first preprocessing procedure is the calculation of the local ridge orientation. The least mean square orientation estimation algorithm [6] is used and the local ridge orientation is specified by blocks rather than every pixel. The calculated orientation is in the range between 0 and π .

After the ridge orientation calculation, ridge frequency is calculated [6]. Using the calculated orientations and frequencies, the input grayscale image is enhanced and binarized by Gabor filters which have both frequency-selective and orientation-selective properties [6].

The final preprocessing operation required before extracting minutiae is thinning. Thinning reduces the widths of the binary ridges down to a single pixel to facilitate the job of detecting ridge endings and bifurcations. The Zhang-Seun thinning algorithm [7] is used in this paper.

2.2 Minutiae Extraction

After a thinned fingerprint image is obtained, minutiae are directly extracted from the thinned image. To detect minutiae, a count of the pixel value transition at a point of interest in a 3×3 mask is used [4], [5]. If the count equals 2, then the point is an endpoint. If the count equals 6, then the point is a bifurcation. For each extracted minutia, the x & y coordinate and the orientation are recorded. The minutiae orientation is defined as the local ridge orientation of the associated ridge [2]. The minutiae orientation is in the range between 0 and π .

3 Proposed Minutiae Postprocessing Algorithm

There are many false minutiae among the extracted minutiae. False minutiae will decrease the performance of the fingerprint identification system by increasing both FRR and FAR. Typical false minutiae structures are shown in Fig. 2. In this section, minutiae postprocessing algorithm is proposed. To eliminate false minutiae without eliminating true minutiae, the proposed methods are based on the flow of ridges as well as the minutiae distance and connectivity.

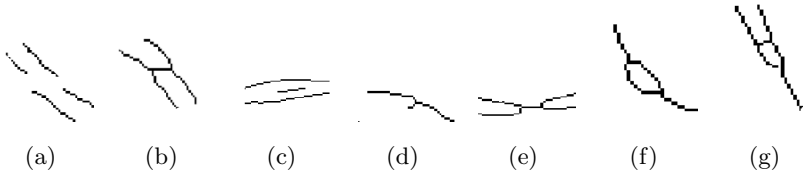


Fig. 2. Typical false minutiae : (a) Broken ridge, (b) Bridge, (c) Short ridge, (d) Short ridge, (e) Short Ridge, (f) Hole, (g) Triangle.

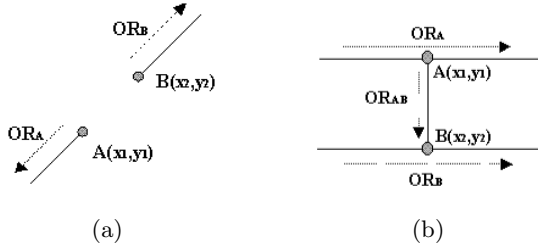


Fig. 3. False Minutiae Structures: (a) Broken ridge, (b) Bridge.

3.1 Detecting Broken Ridge Structure

Because of scars and insufficient finger pressure on the input device, a ridge may break into two ridges creating two endpoints. Obviously, these two endpoints are false minutiae and should be eliminated. Two endpoints are identified as a broken ridge structure by the following decision rules. (Fig 3(a)).

(1)

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} < Dist_1 \quad (1)$$

(2) The line constructed by connecting two endpoints and two ridges connected with each minutia should all flow in the same direction.

$$\tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \simeq \frac{1}{2}(OR_A + OR_B), \quad (2)$$

(3) Two ridges should be flowing to the opposite direction without being connected. For example (Fig 3(a)), if ridge connected with minutia A flows downwards, the other ridge should flow upwards and minutia B should be placed above the minutia A.

3.2 Detecting Bridge Structure

Due to excessive finger pressure or noise in the image, two separate ridges are sometimes connected by a short ridge to make a bridge structure. Based on the fact that ridges in fingerprint flow smooth and neighbor ridges flow in similar direction, method for detecting two false bifurcation associated with bridge structures is as follows. (Fig 3(b)).

- (1) Start tracking three ridges connected to a bifurcation(Point A).
- (2) If one of the tracked ridges meet another bifurcation(Point B), calculate orientation of the ridge connected by two bifurcations(OR_{AB}) and the distance between two bifurcations($Dist_{AB}$).
- (3) If the $Dist_{AB}$ is less than a threshold value($Dist_2$) and the difference between the OR_{AB} and the average orientation of two bifurcations(OR_A, OR_B) is larger than a specified angle($\frac{\pi}{4}$ used in this dissertation), then two bifurcations are identified as a bridge structure.

Note that by applying the described rule, false minutiae in triangular structures (Fig. 2(g)) can also be detected efficiently. Two false minutiae and a true minutia are form a triangular structure. In the triangular structure, it is important to eliminate the two false minutiae while not eliminating the true minutia. By using the rule above, only two false minutiae are detected.

3.3 Detecting Short Ridge Structure

All short ridges should be considered as false minutiae because they are usually artifacts introduced by image preprocessing procedure such as ridge segmentation and thinning. To detect this kind of false minutiae, we start tracking ridges from ridge endings. If a tracked ridge meets another endpoint or a bifurcation within a distance($Dist_3$), two minutiae are considered as false minutiae. Also if a bifurcation meets another bifurcation while tracking ridges and two bifurcations flow in opposite direction(Fig. 2(e)), two bifurcations are considered false minutiae.

3.4 Detecting Hole Structure

Hole structures occur due to pores and dirts on fingerprints. The hole structure can be detected by tracking three ridges connected to an extracted bifurcation. If two tracked ridges meet to form another bifurcation and two bifurcations are within a distance($Dist_4$), then both bifurcations are considered as false minutiae.

3.5 Thresholds and False Minutiae Elimination

Because fingers are elastic, distances between ridges change every time due to different pressure a user puts on a input device. To cope with this problem, various thresholds used in this paper are made adaptive based on ridge frequency (Table. II). Ridge frequency is already calculated in image enhancement process [6].

To efficiently eliminate false minutiae while retaining true minutiae, false minutiae are detected and eliminated in specific order as shown in Fig. 4.



Fig. 4. False Minutiae Elimination Order.

Table 1. Threshold Values : Freq indicates the ridge frequency at the minutia.

<i>Dist</i>	Description	Threshold
<i>Dist</i> ₁	Broken Ridge	2/freq
<i>Dist</i> ₂	Bridge	1.5/freq
<i>Dist</i> ₃	Short Ridge	1.7/freq
<i>Dist</i> ₄	Hole	2/freq

Table 2. Postprocessing performance: Method(A)- Raw Minutiae Extraction, Method(B)-Postprocessing adopted (DMR : Dropped Minutiae Ratio, EMR : Exchanged Minutiae Ratio, TMR : True Minutiae Ratio, FMR : False Minutiae Ratio).

	Method A	Method B
DMR(%)	9.8	12.3
EMR(%)	6.1	5.8
TMR(%)	84.1	81.9
FMR(%)	54.2	21.2

4 Experimental Results

In this section, the performance of the proposed minutiae postprocessing algorithm is evaluated. Fingerprint images were acquired through optic-based fingerprint sensor manufactured by Nitgen. The size of the image is 248×292 with the resolution of 450 dpi and 1000 fingerprint images(10 fingerprints for 100 individuals) with various image qualities were used for experiments.

Before showing the experimental results, we will describe some terms used to evaluate the performance. True Minutiae(TM) are minutiae picked by an expert. Paired Minutiae(PM) are minutiae extracted by the system which coincide with TM. False Minutiae(FM) are minutiae extracted by the system which do not coincide with TM. Dropped Minutiae(DM) are minutiae picked by an expert which are not extracted by the system. Finally, Exchanged Minutiae(EM) are minutiae extracted by the system which coincide with TM except the type.

Table 2 shows the performance of our proposed minutiae postprocessing algorithm. The method A indicates the results of the raw minutiae extraction(without postprocessing) and the method B indicates the results when our postprocessing algorithm is adopted. It shows that the False Minutiae Ratio drops 33% while the True Minutiae Ratio only drops 2.2%.

In addition, to see the effect of our postprocessing algorithm on actual fingerprint matching, we adopted matching procedure from [8]. Fig 5 shows matching results with ROC curves. It is clear from the results that the performance of the matching system is greatly improved by adopting the proposed postprocessing algorithm.

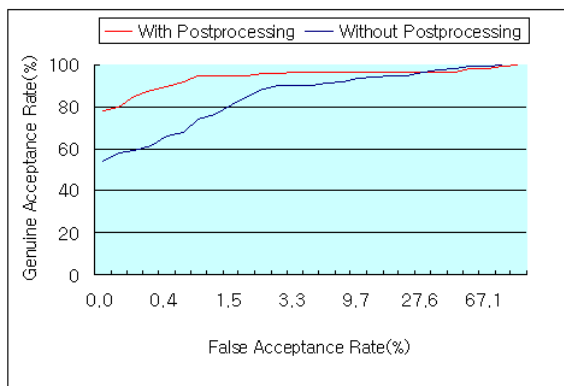


Fig. 5. ROC (Receiver Operating Characteristic) Curves.

5 Conclusion

A minutiae postprocessing algorithm was proposed in this paper. To avoid eliminating true minutiae while postprocessing, our proposed algorithm was based on the orientation and flow of ridges as well as minutiae distance and connectivity. Experimental results showed that our algorithm is indeed very effective; eliminating great deal of false minutiae while retaining most of true minutiae. It was also shown that the proposed algorithm improves the fingerprint matching performance.

References

1. D. Maio and D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 1, pp. 27 - 39, 1997.
2. A. K. Jain, L. Hong, and R. Bolle, "On-Line Fingerprint Verification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp.302-313, April 1997.
3. L. C. Jain et al. Eds., *Intelligent Biometric Techniques in Fingerprint and Face Recognition*, CRC Press International Series on Computational Intelligence, 1999.
4. N. K. Ratha, S. Chen, and A. K. Jain, "Adaptive Flow Orientation Based Feature Extraction in Fingerprint Images," *Pattern Recognition*, vol. 28, no. 11, pp. 1,657 - 1,672, 1995.
5. Q. Xiao and H. Raafat, "Fingerprint Image Postprocessing : A Combined Statistical and Structural Approach," *Pattern Recognition*, vol. 28, no. 11, pp. 1,657 - 1,672, 1995.
6. L. Hong, Y. Wan, and A. K. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.20, pp.777-789, Aug. 1998.
7. J. R. Parker, *Algorithms for Image Processing and Computer Vision*, New York : Wiley Computer Publishing, 1997.
8. A. Wahab, S. H. Chin, and E. C. Tain, "Novel Approach to Automated Fingerprint Recognition," *IEE Proc.- Vis. Image Signal Process*, vol.145, no.3, pp.160-166, Jun. 1998.

Fingerprint Classification by Combination of Flat and Structural Approaches

Gian Luca Marcialis¹, Fabio Roli¹, and Paolo Frasconi²

¹Dept. of Electrical and Electronic Eng., University of Cagliari
Piazza d Armi, I-09123 Cagliari, Italy
{marcialis, roli}@diee.unica.it

²Dept. of Systems and Computer Science, University of Florence
Via di Santa Marta 3, I-50139, Florence, Italy
paolo@dsi.unifi.it

Abstract. This paper investigates the advantages of the combination of flat and structural approaches for fingerprint classification. A novel structural classification method is described and compared with the multichannel flat method recently proposed by Jain et al. [1]. Performances and complementarity of the two methods are evaluated using NIST-4 Database. A simple approach based on the concept of metaclassification is proposed for the combination of the two fingerprint classification methods. Reported results point out the potential advantages of the combination of flat and structural fingerprint-classification approaches. In particular, such results show that the exploitation of structural information allows increasing classification performances.

1. Introduction

The huge size of fingerprint databases used for real applications seriously affects the identification time of AFISs (Automated Fingerprint Identification Systems). Automatic fingerprint classification, based on well known schemes of fingerprint subdivision into classes (e.g., the Henry's classification scheme [2]), is the usual strategy adopted for reducing the number of comparisons during the fingerprint identification process and, consequently, for reducing the identification time [3]. Many approaches to fingerprint classification have been presented in the literature and this research topic is still very active. Overviews of the literature can be found in [1], [4]. In our opinion, the proposed approaches to automatic fingerprint classification can be coarsely subdivided into the two main categories of flat and structural approaches. Flat approaches are characterised by the use of the decision-theoretic (or statistical) approach to pattern classification, namely, a set of characteristic measurements, called feature vector, is extracted from fingerprint images and used for classification [1]. On the other hand, structural approaches presented in the literature basically use the syntactic or structural pattern-recognition methods [4]. Fingerprints are described by production rules or relational graphs and parsing processes or graph

matching algorithms are used for classification. It is worth remarking that the structural approach to fingerprint classification has not received much attention until now. However, a simple visual analysis of the structure of fingerprint images allows one to see that structural information can be very useful for distinguishing some fingerprint classes (e.g., for distinguishing fingerprints belonging to class A from the ones of class W [4]). On the other hand, it is easy to see that structural information is not appropriate for distinguishing fingerprints of the L, R and T classes [4]. Accordingly, the combination of flat and structural approaches should be investigated. With regard to this issue, it is worth noting that, to the best of our knowledge, very few papers investigated the potentialities of such combination [11].

In this paper the advantages of the combination of flat and structural approaches for fingerprint classification are investigated. A novel structural approach is described and compared with the multichannel flat method recently proposed by Jain et al. [1]. The advantages of combining these two approaches are then investigated by experiments.

2. Combination of Flat and Structural Fingerprint Classification Approaches

2.1. The Proposed Structural Approach to Fingerprint Classification

2.1.1. An Overview of the Proposed Approach

It is quite easy to see by a visual analysis of fingerprint images that fingerprint structure can be extracted by segmenting fingerprint into regions characterized by homogeneous ridge directions. Therefore, we propose to extract and represent the structural information of fingerprints by segmenting the related directional images and by converting such segmented images into relational graphs whose nodes correspond to regions extracted by segmentation algorithm. Graph nodes are then characterized by local characteristics of regions and by the geometrical and spectral relations among adjacent regions.

In particular, the main steps of our structural approach to fingerprint classification are as follows:

- 1) Computation of the directional image of the fingerprint. This directional image is a 28×30 matrix. Each matrix element represents the ridge orientation within a given block of the input image. Such computation of the directional image is performed using the method proposed in [5].
- 2) Segmentation of the directional image into regions containing ridges with similar orientations. To this end, the segmentation algorithm described in [6] was used.
- 3) Representation of the segmented image by a D.O.A.G (Directed Oriented Acyclic Graph). The representation of fingerprint structure is completed by characterizing each graph node with a numerical feature vector.
- 4) Classification of the above DOAG by a RNN (Recursive Neural Networks) made up of two multilayer perceptrons (MLPs) neural nets. This neural network model is briefly described in section 2.1.3. Details can be found in [7].

2.1.2. Image Segmentation and Relational Graph Construction

In order to segment directional fingerprint images, we used an algorithm explicitly designed for such task [6].

The construction of relational graphs from segmented images is performed by the following main steps:

- the image region containing the core point is selected as starting region for the graph construction, that is, a graph node associated to such region is initially created;
- the regions that are adjacent to such core region are then evaluated for the creation of new nodes. Nodes are created for adjacent regions which are located in one of the following spatial positions with respect to the core region: North, North East, East, South East, South, South West, West, North West;
- the above process is repeated for each of the new nodes until that all the regions of the segmented images have been considered;
- the graph nodes created by the above algorithm are finally characterized by a numerical feature vector containing local characteristics of the related image regions (area, average directional value, etc) and some geometrical and spectral relations with respect to adjacent regions (relative positions, differences among directional average values, etc).

2.1.3. A Neural Network Model for Structural Fingerprint Classification

Our approach will rely on Recursive Neural Network [7], a machine learning architecture which is capable of learning to classify hierarchical data structures, such as the structural representation of fingerprints which we employ in this paper. The input to the network is a labeled DOAG U , where the label $U(v)$ at each vertex v is a real-valued feature vector associated with a fingerprint region, as described in Section 2.1.2. A hidden state vector $X(v) \in \mathcal{R}^n$ is associated with each node v , and this vector contains distributed representation of the subgraph dominated by v (i.e., all the vertices that can be reached starting a directed path from v). The state vector is computed by a state transition function f which combines the state vectors of v 's children with a vector encoding of the label of v . Computation proceeds recursively from the frontier to the supersource (the vertex dominating all other vertices). The base step of such computation is $X(v) = 0$ if v is a missing child. Transition function f is computed by a multilayer perceptron, which is replicated at each node in the DOAG, sharing weights among replicas. Classification with recurrent neural networks is performed by adding an output function g that takes as input the hidden state vector $X(s)$ associated with the supersource s . Function g is also implemented by a multilayer perceptron. The output layer in this case uses the *softmax* functions (normalized exponentials), so that Y can be interpreted as a vector of conditional probabilities of classes given the input graph, i.e. $Y_i = P(C = i | U)$, being C a multinomial class variable. Training relies on maximum likelihood.

2.2. Combination of Flat and Structural Approaches

In order to investigate the potentialities of the combination of flat and structural methods, we coupled our structural approach with a flat approach recently proposed by Jain et al. [1]. The core of such flat approach is a novel representation scheme (called FingerCode) which is able to represent into a numerical feature vector both the minute details and the global ridge and furrows structures of fingerprints. Several strategies for combining classifiers were evaluated in order to combine the flat and structural approaches [8-10]. We firstly assessed the performances of simple combination rules, namely, the majority voting rule and the linear combination, which require the assumption of error independence among the combined classifiers. Such combination rules performed poorly due to the strong error correlation between the flat and the structural classifiers considered. (It is worth noting that the two classifiers make a lot of identical errors on NIST-4 database). Accordingly, we adopted the so-called metaclassification (or stacked) approach to classifier combination which uses an additional classifier for combination [10]. In particular, a K-nearest neighbour classifier was used.

3. Experimental Results

3.1. The Data Set

The NIST-4 database containing five fingerprint classes (A, L, R, W, T) was used for experiments. In particular, the first 1,800 fingerprints (f0001 through f0900 and s0001 through s0900) were used for classifier training. The next 200 fingerprints were used as validation set, and the last 2,000 fingerprints as test set.

3.2. Performances and Complementarity of Flat and Structural Classifiers

First of all, we assessed the performances of the flat and structural approaches described in Section 2. With regard to the multichannel flat approach, a multilayer perceptron (MLP) using the FingerCode feature vector as input was trained and tested on NIST-4 database. The best performances on the test set were obtained with a MLP architecture with 28 hidden units. Table 1 shows such performances.

With regard to the structural approach described in Section 2.1, Table 2 shows the confusion matrix on the NIST-4 test set of the related recursive neural network.

Tables 1 and 2 point out that the accuracy of the structural classifier is much lower than the one of the flat classifier. As pointed out in Section 1, this is mainly due to the large degree of confusion among L, R and T classes. On the other hand, as expected, the best performances of the structural classifier are related to the discrimination between A and W classes.

Table 1. Confusion matrix on the NIST-4 test set of the multilayer perceptron neural network. Percentage accuracy values are given in the table. The FingerCode feature vector was used as input of this network.

	A	L	R	T	W
A	80.51	0.93	1.86	16.71	0.00
L	1.58	91.84	0.53	3.95	2.11
R	1.54	0.00	89.49	6.92	2.05
T	14.91	3.25	2.17	79.13	0.54
W	1.01	3.28	6.06	0.25	89.39
Overall accuracy 86.01%					

Table 2. Confusion matrix on the NIST-4 test set of the recursive neural network. Percentage accuracy values are given in the table.

	A	L	R	T	W
A	76.64	2.57	3.27	16.36	1.17
L	2.09	67.62	2.61	15.93	11.75
R	2.42	4.12	73.12	14.04	6.30
T	12.98	14.45	13.86	51.92	6.78
W	1.02	7.89	6.36	0.00	84.73
Overall Accuracy: 71.47%					

Afterwards, we analyzed the degree of complementarity between the two above classifiers. To this end, the performances of an ideal oracle that, for each input fingerprint, always selects the one of two classifiers, if any, that classifies correctly such fingerprint were assessed. Such oracle applied to the two above classifiers provided an overall accuracy on NIST-4 test set of 92.54%. This accuracy value obviously represents a very tight upper bound for any combination method applied to the two classifiers. However, it points out the potential benefit of the combination of the flat and structural classifiers.

3.3. Combination of Flat and Structural Classifiers

As described in Section 2.2, a K-nearest neighbour classifier (with a value of the *k* parameter equal to 113) was used for combining the flat and structural classifiers. Such metaclassifier takes the outputs of the two above classifiers as inputs and provides the final fingerprint classification as output. Table 3 depicts the confusion matrix on the NIST-4 test set of the combination of the flat and structural classifiers. It is worth noting that such combination outperforms the best single classifier (i.e., the MLP classifier using the FingerCode representation; see Table 1), so pointing out that the exploitation of structural information allows increasing classification performances. In particular, we remark that, as expected (section 1), such combination improves the performances related to A and W classes.

Table 3. Confusion matrix on the NIST-4 test set of the combination of the flat and structural classifiers.

	A	L	R	T	W
A	83.95	0.00	0.93	15.12	0.00
L	1.35	92.18	0.54	4.85	1.08
R	1.81	0.00	89.15	7.49	1.55
T	12.27	2.40	2.13	83.20	0.00
W	0.76	2.54	4.83	0.51	91.35
Total Accuracy: 87.88%					

Acknowledgements

The authors wish to thank Anil K. Jain for providing them with the FingerCode representation of NIST-4 data set. A special thank to Salil Prabhakar who helped the authors to understand and use the FingerCode representation. The authors also wish to thank Raffaele Cappelli and Davide Maltoni for providing them with the results of the application of their image segmentation algorithm to NIST-4 data set.

References

- [1] A.K. Jain, S. Prabhakar, and L. Hong, A Multichannel Approach to Fingerprint Classification , IEEE Transactions on PAMI, vol.21, no.4, pp. 348-358, 1999.
- [2] E.R. Henry, Classification and Uses of Fingerprints, Routledge, London (1900).
- [3] Biometrics - Personal Identification in Networked Society, Kluwer Academic Publishers, A.K. Jain, R. Bolle and S. Pankanti Editors, 1999.
- [4] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, Fingerprint Classification by Directional Image Partitioning , IEEE Trans. on PAMI, vol.21, no.5, pp. 402-421, 1999.
- [5] G.T. Candela et al., "PCASYS - A Pattern-Level Classification Automation System for Fingerprints", NIST tech. Report NISTIR 5647, 1995.
- [6] D. Maio and D. Maltoni, "A Structural Approach to Fingerprint Classification", Proc. 13th ICPR, Vienna, 1996, pp. 578-585.
- [7] P. Frasconi, M. Gori, and A. Sperduti, A General Framework for Adaptive Processing of Data Structures , IEEE Trans. On Neural Networks, vol.9, no.5, pp.768-786, 1998.
- [8] G. Giacinto, F. Roli, and L. Bruzzone, "Combination of Neural and Statistical Algorithms for Supervised Classification of Remote-Sensing Images", Pattern Recognition Letters, May 2000, vol. 21, no. 5, pp. 385-397.
- [9] Kittler, J. and Roli, F.: Proc. of the First International Workshop on Multiple Classifier Systems (MCS 2000). Springer-Verlag Pub., Lecture Notes in Computer Science, Vol. 1857, (2000) pp. 1-404.
- [10] G. Giacinto and F. Roli, "Ensembles of Neural Networks for Soft Classification of Remote Sensing Images", European Symposium on Intelligent Techniques, 20-21 March, 1997, Bari, Italy, pp. 166-170.
- [11] R. Cappelli, D. Maio, and D. Maltoni, Combining Fingerprint Classifiers , Proc. of the First International Workshop on Multiple Classifier Systems (MCS 2000). Springer-Verlag Pub., Lecture Notes in Computer Science, Vol. 1857, (2000), pp. 351-361.

Using Linear Symmetry Features as a Pre-processing Step for Fingerprint Images

Kenneth Nilsson and Josef Bigun

Halmstad University, School of Information Science
Computer and Electrical Engineering,
Box 823, S-301 18 Halmstad, Sweden

{[@ide.hh.se](mailto:Kenneth.Nilsson,Josef.Bigun)}

Abstract. This paper presents the idea to use linear symmetry properties as a feature based pre-processing step for fingerprint images. These features contain structural information of the local patterns. The linear symmetry can be computed by using separable spatial filtering and therefore has the potential to be a fast pre-processing step. Our results indicate that minutiae can be located as well as can be assigned a certain class type. The type of minutiae matching in combination with geometrical matching increases the matching efficiency as compared to the pure geometrical matching.

1 Introduction

In person identification systems using fingerprint images minutiae and their relative positions to each other are often used in the matching process. Minutiae are found in positions where the simple pattern in the fingerprint image is changed. In this position an identifiable signature, e.g. a ridge is suddenly broken, two ridges merge, etc. is seen. When the minutiae coordinates are extracted an alignment of the minutiae point-pattern to a reference is done (matching process). The entire matching process including processing of images may be inefficient due to false minutiae as well as missing correct minutiae. Also different minutiae point-patterns are of different sizes [1]. A review of finger-print identification studies is given in [1].

The aim of this work is to investigate if the local linear symmetry (LS) property can be used as a pre-processing step for fingerprint images in an attempt to increase the efficiency of the minutiae matching process. The pre-processing step should locate, and also assign a feature vector to each found minutiae point. Using a feature vector gives additional information compared to only have the locations of the minutiae. Also, representing a minutiae point by a feature vector it can be classified belonging to a specific type of minutiae. Knowing also the type of minutiae, the later matching process can be done more effectively.

It is desirable to find a representation of a fingerprint image that combines both local and global information [2]. In our representation the feature vector has the local

information and the relative positions of the found minutiae points have the global information.

2 Linear Symmetry

A local neighborhood where the gray value only changes in one direction has linear symmetry, LS. A linear symmetry property defines also an orientation. Mathematically, linear symmetry orientation is the direction along which an infinitesimal translation leaves the pattern least variant, (ideally invariant). A fully equivalent formulation is that the linear symmetry orientation is the least inertia axis orientation of the Fourier transform energy of the pattern (ideally concentrated to a line, to which this symmetry originally referred to). The LS property can be extracted by using local spatial filtering [3] via separable gaussian filters and gaussian derivative filters. The LS property is obtained by applying averaging to the LS tensor which corresponds to a physical property, a vector field representing local orientations or the lack of local orientations, which does not change with the observer (coordinates). The LS vector field is three dimensional consisting of 3 real numbers when the images are 2-D (e.g. finger prints). In this case these 3 real numbers can be conveniently represented as one complex number, called I_{20} , and one real number, called I_{11} . The complex number is obtained via $I_{20} = \langle \text{Grad}(f), \text{Grad}^*(f) \rangle$ where $\text{Grad} = D_x + i D_y$ is an operator, and the \langle, \rangle is the usual scalar product, having a gaussian as a kernel. The real number is obtained via $I_{11} = \langle |\text{Grad}(f)|, |\text{Grad}^*(f)| \rangle$. The LS property can be generalized to comprise deformations of the local image, which allows to detect singular points such as spiral centers, [4]. In this case however, the definition of the Grad operator is changed to a gradient consisting of Lie derivatives, and the averaging kernel is replaced by a harmonic function multiplied by a gaussian.

2.1 Pre-processing

The output from the LS pre-processing step are two images I_{20} and I_{11} . I_{20} is displayed as a vector image (figure 1) where each pixel is represented by a vector. The length of the vector is a measure of the local LS strength and the argument is the estimated local orientation. I_{11} is a scalar image, it measures the amount of local gray value change. It can be shown that inequality $|I_{20}| \leq I_{11}$ holds, and that the upper bound of $|I_{20}|$ is attainable (equality holds) if and only if the pattern possesses LS property, an infinitesimal translation in a certain direction yields zero error.

2.2 Extraction and Labeling of Minutiae

Minutiae are found by the lack of LS. Here the lack of LS is computed as $1 - |I_{20}|/I_{11}$, which is never negative. This takes values in the interval $[0, 1]$, where a high value indicates the lack of LS. Besides that, each minutiae is represented by a feature vector.

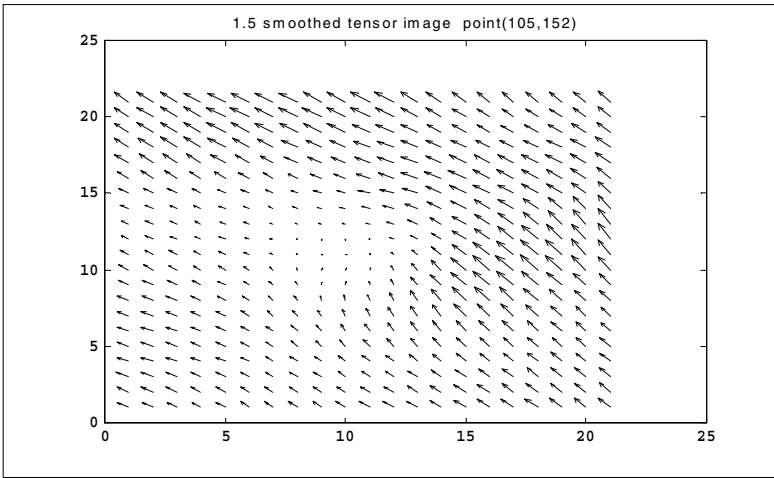


Fig. 1. The vector image I_{20}

As features the LS property in the *neighborhood* of the minutiae is used. To get a translation and rotation invariant numeric representation, the magnitude of the 1D Fourier transform of the extracted features is calculated. This will be described further below.

For use in a later matching process each minutiae is labeled as belonging to a specific class. Within each class there should be similar linear symmetry properties of the neighborhood of the minutiae. The classes are found by clustering, and the labeling of minutiae is done by finding the closest class (minimum distance to the cluster centers).

3 Experiments

The fingerprint images used are from the FVC2000 database, DB2 set A, and are captured by using a low cost capacitive sensor. The size of an image is 364 x 256 pixels and the resolution is 500 dpi.

In computing the I_{20} image the first partial derivatives of the 2D gaussian function are used. The size of the derivative filters are 1 x 11 with $\sigma=1.8$, and the smoothing filter, also gaussian, is 1 x 13, with $\sigma=2.2$. Figure 1 shows the vector image I_{20} in a neighborhood of a minutiae.

3.1 Extraction of Minutiae

First a low pass filtering of the fingerprint image is applied, and also to favor pixels in the center of the image the low pass filtered image is multiplied by a 2D gaussian.

From the lack of LS image the 15 strongest minutiae are extracted. When a minutiae is found, coordinates close to it are occluded in the search for new minutiae.

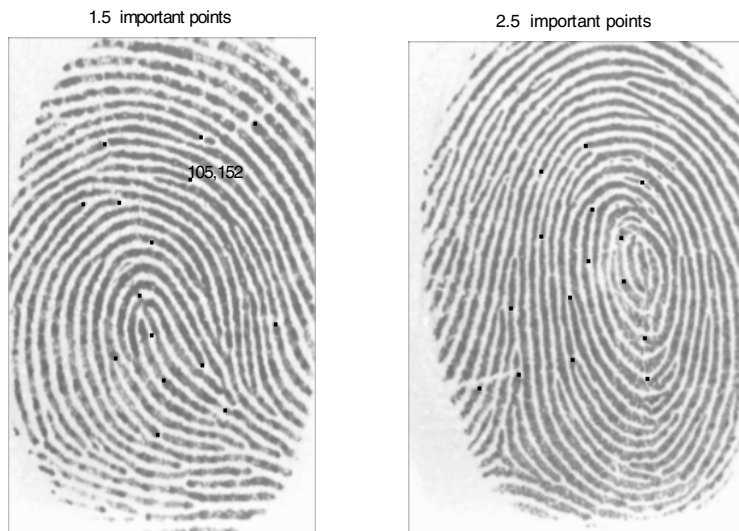


Fig. 2. Marked minutiae points

3.2 Feature Extraction

The LS property on a circle around each minutiae point are extracted. In particular, the vector image I_{20} is sampled in $N=16$ equally spaced number of points on a circle with a radius of 15 pixels.

To get a translation and rotation invariant representation of the linear symmetry property, and also to get a more compressed representation, the magnitude of the 1D Fourier transform of the 16 extracted points is calculated.

Also, only the lowest Fourier coefficients are used in the feature vector representing the neighborhood of the minutiae.

3.3 Labeling of Minutiae

For use in a later matching process each minutiae is labeled belonging to a specific class. Each specific class is represented by its feature vector (class-center). Minutiae within a class should have similar LS properties.

To find the specific classes and their class-centers, 100 images from different persons are used. From each of the 100 images 15 minutiae are extracted and their feature vectors are calculated. The first six Fourier coefficients are used as the feature vector representing the minutiae point.

The class-centers are calculated by a clustering technique. The fuzzy c-means clustering algorithm, [5] is utilized. Minutiae are labeled according to the shortest Euclidian distance to a class-center. The number of classes is 10.

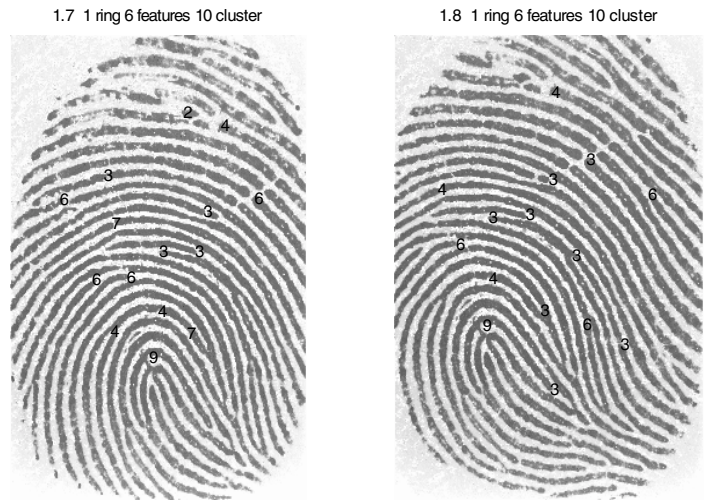


Fig. 3. Labeled minutiae points, person 1

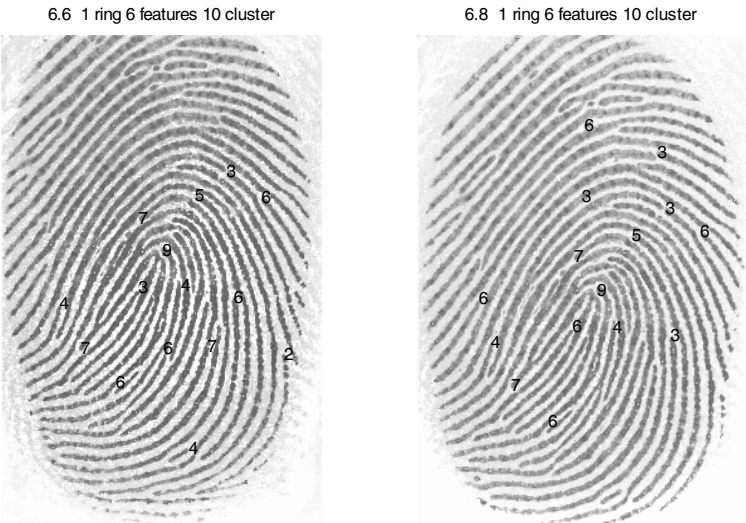


Fig. 4. Labeled minutiae points, person 6

Figure 3 shows fingerprint images from a person captured at different times. Figure 4 is the fingerprints from another person.

Most of the common minutiae are labeled with the same class-label (in figure 3: 7 of 9, in figure 4: 9 of 11), which indicates that the LS property has enough information to be used in a pre-processing step.

4 Conclusion

Our experiments indicate that the LS property has the necessary information for both extracting the minutiae points and also labeling each minutiae belonging to a specific type. More experiments have to be done on fingerprint images with differences in quality.

Acknowledgement

This work has been supported by the national VISIT program.

References

1. Jain, A.K., Hong, L., Pankanti, S., Bolle, R.: An identity authentication system using fingerprints. *Proceedings of the IEEE*, Vol. 85, No. 9, September 1997, 1365-1388
2. Jain, A.K., Prabhakar, S., Hong, L., Pankanti, S.: Filterbank based fingerprint matching. *IEEE Transactions on Image Processing*, Vol. 9, No. 5, May 2000, 846-859
3. Bigun, J., Granlund, G.H.: Optimal orientation detection of linear symmetry. *First International Conference on Computer Vision*, London, June 1987, 433-438
4. Bigun, J.: Pattern recognition in images by symmetries and coordinate transformations. *Computer Vision and Image Understanding*, Vol. 68, No 3. December 1997, 290-307
5. Bezdek, J.C.: *Pattern recognition with fuzzy objective function algorithm*. Plenum Press, 1981

Fingerprint Classification with Combinations of Support Vector Machines

Yuan Yao¹, Paolo Frasconi², and Massimiliano Pontil^{3,1}

¹ Department of Mathematics, City University of Hong Kong, Hong Kong

² Department of Systems and Computer Science, University of Firenze, Firenze, Italy

³ Department of Information Engineering, University of Siena, Siena, Italy

Abstract. We report about some experiments on the fingerprint database NIST-4 using different combinations of Support Vector Machine (SVM) classifiers. Images have been preprocessed using the feature extraction technique as in [11]. Our best classification accuracy is 89.3 percent (with 1.8 percent rejection due to the feature extraction process) and is obtained by an error-correction scheme of SVM classifiers. Our current system does not outperform previously proposed classification methods, but the focus here is on the development of novel algorithmic ideas. In particular, as far as we know, SVM have not been applied before in this area and our preliminary findings clearly suggest that they are an effective and promising approach for fingerprint classification.

1 Introduction

The pattern recognition problem studied in this paper consists of classifying fingerprint images into one out of five categories: whorl (W), right loop (R), left loop (L), arch (A), and tented arch (T). These categories were defined during early investigations about fingerprint structure [9] and have been used extensively since then. The task is interesting because classification can be employed as a preliminary step for reducing complexity of database search in the problem of automatic fingerprint matching [7, 11]. Basically, if a query image can be classified with high accuracy, the subsequent matching algorithm only needs to compare stored images belonging to the same class.

Several pattern recognition algorithms have been proposed for fingerprint classification, including early syntactic approaches [12], methods based on detection of singular points [1], connectionist algorithms such as self-organizing feature maps [8], neural networks [13], and structural methods based on (dynamic) graph matching [1]. The current highest accuracy (92.2 percent on the NIST Database 4) was obtained with a method based on multi-space principal component analysis [2]. In spite of these efforts, the problem has not been solved satisfactorily and there is undoubtedly room for further improvements in terms of classification accuracy, rejection threshold, and simplicity of design.

In this paper, we propose an fingerprint classifier based on Support Vector Machines (SVM), a relatively new technique to train classifiers that is well-founded in statistical learning theory [14]. One of the main attractions of using SVMs is that they are capable of learning in *sparse, high-dimensional spaces*

with very few training examples. SVMs have been successfully applied to various classification problems (see [3] and references therein).

Since basic SVM are formulated for solving binary classification tasks, we designed a multiclass strategy based on a new error correcting code (ECC) strategy.

Our system is validated on FingerCode [10] preprocessed images from the NIST database 4 [18]. The best SVM combination achieves 89.3 percent accuracy with 1.8 percent rejection, due to failures of FingerCode in reliably locating the fingerprint core. This result is only 0.7 percent worse than the accuracy obtained in [10] using the same features and a two stages k -NN/MLP classifier. Interestingly, SVM's accuracy is much better than separate accuracies of both k -NN and MLP, but slightly worse than the cascading of the two. Hence, although preliminary, we believe our results are very promising and might yield state-of-the-art improvements if further refined.

2 Support Vector Machines

Support vector machines (SVMs) [17] perform pattern recognition for two-class problems by determining the separating hyperplane¹ with maximum distance to the closest points of the training set. These points are called *support vectors*. If the data is not linearly separable in the input space, a non-linear transformation $\Phi(\cdot)$ can be applied which maps the data points $\mathbf{x} \in \mathbb{R}^n$ into a high (possibly infinite) dimensional space \mathcal{H} which is called feature space. The data in the feature space is then separated by the optimal hyperplane as described above.

The mapping $\Phi(\cdot)$ is represented in the SVM classifier by a kernel function $K(\cdot, \cdot)$ which defines an inner product in \mathcal{H} , i.e. $K(\mathbf{x}, \mathbf{t}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{t})$. The decision function of the SVM has the form:

$$f(\mathbf{x}) = \sum_{i=1}^{\ell} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}), \quad (1)$$

where ℓ is the number of data points, and $y_i \in \{-1, 1\}$ is the class label of training point \mathbf{x}_i . Coefficients α_i in Eq. (1) can be found by solving a quadratic programming problem with linear constraints [17]. The support vectors are the nearest points to the separating boundary and are the only ones for which α_i in Eq. (1) can be nonzero.

An important family of admissible kernel functions are the Gaussian kernel, $K(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|/2\sigma^2)$, with σ the variance of the gaussian, and the polynomial kernels, $K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x} \cdot \mathbf{y})^d$, with d the degree of the polynomial. For other important examples of kernel functions used in practice see [17].

Let M be the distance of the support vectors to the hyperplane. This quantity is called *margin* and it is related to the coefficients in Eq. (1),

$$M = \left(\sum_{i=1}^{\ell} \alpha_i \right)^{\frac{1}{2}}. \quad (2)$$

¹ SVM theory also includes the case of non-separable data, see [17].

The margin is an indicator of the separability of the data. In fact, the expected error probability of an SVM is bounded by the average (with respect to the training set) of $\frac{R^2}{\ell M^2}$, with R the radius of the smallest sphere containing the data points in the feature space.

2.1 Multi-class Classification

Many real-world classification problems involve more than two classes. Attempts to solve q -class problems with SVMs have involved training q SVMs, each of which separates a single class from all remaining classes [17], or training q^2 machines, each of which separates a pair of classes [2, 6, 15]. The first type of classifiers are usually called *one-vs-all*, while classifiers of the second type are called *pairwise* classifiers. For the one-vs-all a test point is classified into the class whose associated classifier has the highest score among all classifiers. In the pairwise classifier, a test point is classified in the class which gets most votes among all the possible classifiers [6].

Classification schemes based on training one-vs-all and pairwise classifiers are two extreme approaches: the first uses all the data, the second the smallest portion of the data. In practice, it can be more effective to use intermediate classification strategies in the style of error-correcting codes [10, 16]. In this case, the number of classifiers grows linearly with the number of classes. Each classifier is trained to separate a subset of classes from another disjoint subset of classes (the union of these two subsets does not need to cover all the classes). For example the first set could be classes A and T and the second classes R, L and W. By doing so, we associate each class with a row of the “coding matrix” $M \in \{-1, 0, 1\}^{q \times s}$, where s denotes the number of classifiers. $M_{ij} = -1$ or 1 means that points in class i are regarded as negative or positive examples for training the classifier j . $M_{ij} = 0$ says that points in class i are not used for training classifier j . A test point is classified in the class whose row in the coding matrix has minimum distance to the output row of the classifiers. The simplest and most commonly used distance is the hamming distance. We will discuss other distance measures in Section 3.

3 Experimental Results

3.1 Dataset

Our system was validated on FingerCode preprocessed fingerprints from the NIST Database 4 [18]. FingerCode is a representation scheme described in [10] and consists of a vector of 192 real features computed in three steps. First, the fingerprint core and center are located. Then the algorithm separates the number of ridges present in four directions (0° , 45° , 90° , and 135°) by filtering the central part of a fingerprint with a bank of Gabor filters. Finally, standard deviations of grayscale values are computed on 48 disc sectors, for each of the four directions. The NIST Database 4 consists of 4000 images analyzed by a human expert and labeled with one or more of the five structural classes W, R, L, A, and T (more

than one class is assigned in cases where ambiguity could not be resolved by the human expert).

Previous works on the same dataset either rejected ambiguous examples in the training set (loosing in this way part of the training data), or used the first label as a target (potentially introducing output noise). The error correcting code developed in this paper allows a more accurate use of ambiguous examples, since each SVM is only in charge of generating one codebit, whose value discriminates between two disjoint sets of classes. If a fingerprint has labels all belonging to the same set for a particular codebit, then clearly we can keep this example in the training set without introducing any labeling noise. As far as testing is concerned, we followed the customary convention of counting as errors only those predictions that do not agree with any of the labels assigned to the fingerprint.

Before discussing our results, we briefly summarize the results in [10]. Three different experiments were performed there: (a) A k -NN classifier with $k = 10$, (b) A MLP classifier, (c) A hybrid combination of (a) and (b): given a test point, first the k -NN classifier is used to compute the two most frequent labels. Then, the MLP corresponding to these two labels is used for the final classification. The accuracies obtained were of 85.4, 86.4, and 90.0 percent, respectively.

3.2 Results with SVMs

We used the three types of multi-class classification schemes discussed in section 2.1 which are based on the combination of binary SVMs. SVMs have been trained using the SVMFu code [1] on a 550MHz Pentium-II PC. Training on 2000 examples takes about 10s for pairwise classifiers and 20s for one-vs-all classifiers.

One-vs-All SVMs. We trained five one-vs-all SVM classifiers using both Gaussian kernels and polynomials of degree between 2 and 6. The best result was obtained with the Gaussian kernel ($\sigma = 1$): 88.0% (see the confusion matrix in Table 2a). The best polynomial SVM was of degree 3 and achieved a performance of 84.5%. Rejection can be decided by examining the margin (see Eq. (2)). Table 1 shows the accuracy-rejection tradeoff obtained in our experiments.

Table 1. Accuracy vs. rejection rate for the one-vs-all SVMs combination.

Rejection Rate:	1.8%	3.0%	6.1%	10.9%	17.3%	24.2%	30.2%
Accuracy:	88.0%	89.0%	89.7%	90.9%	92.4%	93.4%	94.8%

Finally, in the four classes task (classes A and T merged together) a Gaussian SVM with $\sigma = 1$ and $C = 10$ obtains an accuracy of 93.1%. For comparison, the accuracy reported in [10] for the sole MPL's is 92.1%, but the cascade of k -NN and MLP yields 94.8%.

² This software can be downloaded at <http://five-percent-nation.mit.edu/SvmFu>.

Table 2. Confusion matrices: (a) One-vs-all SVMs; (b) Pairwise SVMs; (c) ECC SVMs with margin weighted Euclidean decoding). Rows denote the true class, columns the assigned class.

	W	R	L	A	T
W	356	23	14	3	1
R	4	344	1	7	33
L	4	2	356	6	13
A	0	2	5	371	55
T	0	7	7	48	303

(a)

	W	R	L	A	T
W	359	24	15	3	1
R	4	341	1	6	36
L	5	0	356	6	15
A	0	2	4	363	58
T	0	7	9	38	318

(b)

	W	R	L	A	T
W	362	22	11	3	0
R	4	350	3	8	27
L	7	2	357	5	11
A	0	3	3	398	32
T	0	10	9	51	287

(c)

Pairwise SVMs. We trained the ten pairwise SVMs using always a Gaussian kernels with $\sigma = 1$. The test set accuracy increases to 88.4%, improving of 2% the MLP accuracy reported in [11]. The confusion matrix is reported in Table 2b.

Error-Correction SVM Scheme. Three sets of SVM classifiers were used to construct the coding matrix: 5 one-vs-all classifiers, 10 two-vs-three classifiers and 10 pairwise classifiers. Three kinds of decoding distances were compared: (i) Hamming decoding: 88.0%, (ii) The loss-based decoding proposed in [16]: 88.8%; (iii) Margin weighted decoding (the distance function is defined as the Euclidean distance weighted by the margin): 89.3%. The confusion matrix for the last case is reported in Table 2c.

We have measured the margin as in Eq. (2) and the number of support vectors of each SVM classifier used in our experiments (the training error of each individual classifier was always zero). The number of support vector ranges between 1/5 and 1/2 of the number of training points. As expected the margin decreases for those classifiers which involve difficult pairs of classes. Among the pairwise classifiers, the A-T classifier has the smallest margin. The margin of the T-vs-all and A-vs-all is also small. However the margin of the AT-vs-RLW classifier increases, which might explain why our error correcting strategy works well.

4 Conclusions

We have presented experiments for fingerprint classification using different combinations of SVM classifiers. Images have been preprocessed using the features extraction technique as in [11]. Our best classification accuracy is obtained by an error-correction schemes of SVM classifiers. It improves separate accuracies of both k -NN and MLP of 3.9 and 2.9 percent respectively, while is only 0.7 percent worse than the best performance obtained with the same features [11].

As far as we know, this is the first experimental study of SVMs in the area of fingerprint classification. Therefore, we believe that our preliminary findings are promising and might yield state-of-the-art improvements if further refined. In particular this might be obtained by reweighting the features used by each SVM classifier using the technique in [19].

Acknowledgment: We wish to thank Anil Jain for providing us the dataset of preprocessed NIST-4 fingerprints.

References

1. R. Cappelli, A. Lumini, D. Maio, and D. Maltoni. Fingerprint classification by directional image partitioning. *Transactions on Pattern Analysis Machine Intelligence*, 21(5):402–421, 1999.
2. R. Cappelli, D. Maio, and D. Maltoni. Fingerprint classification based on multi-space KL. In *Proceedings Workshop on Automatic Identification Advances Technologies (AutoID'99)*, pages 117–120, 1999.
3. N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines (and other kernel-based learning methods)*. Cambridge University Press, 2000.
4. T. G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 1995.
5. T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13:1–50, 2000.
6. Jerome H. Friedman. Another approach to polychotomous classification. Technical report, Department of Statistics, Stanford University, 1997.
7. R.S. Germain, A. Califano, and S. Colville. Fingerprint matching using transformation parameter clustering. *IEEE Computational Science and Engineering*, 4(4):42–49, 1997.
8. U. Halici and G. Ongun. Fingerprint classification through self-organizing feature maps modified to treat uncertainty. *Proceedings of the IEEE*, 84(10):1497–1512, 1996.
9. E.R. Henry. *Classification and Uses of Finger Prints*. Routledge, London, 1900.
10. A.K. Jain, S. Prabhakar, and L. Hong. A multichannel approach to fingerprint classification. *PAMI*, 21 (4):348–359, 1999.
11. K. Karu and A.K. Jain. Fingerprint classification. *Pattern Recognition*, 29(3):389–404, 1996.
12. B. Moayer and K.S. Fu. A syntactic approach to fingerprint pattern recognition. *Pattern Recognition*, 7:1–23, 1975.
13. K. Moscinska and G. Tyma. Neural network based fingerprint classification. In *Third International Conference on Neural Networks*, pages 229–232, 1993.
14. J. Platt, N. Cristianini, and J. Shawe-Taylor. Lrge margin dags for multiclass classification. In *Advances in Neural Information Processing Systems*, Denver, Colorado, 2000.
15. M. Pontil and A. Verri. Support vector machines for 3-d object recognition. *IEEE Trans. PAMI*, pages 637–646, 1998.
16. Robert E. Schapire, Yoram Singer, and Erin Lee Young. Reducing multiclass to binary: A unifying approach for margin classifiers. Technical report, AT&T Research, 2000.
17. V. N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.
18. C.I. Watson and C.L. Wilson. National Institute of Standards and Technology, March 1992.
19. J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik. Feature selection for support vector machines. In *NIPS-13*, 2001. To Appear.

Performance Evaluation of an Automatic Fingerprint Classification Algorithm Adapted to a Vucetich Based Classification System

Alberto Bartesaghi, Alicia Fernández, and Alvaro Gómez

Instituto de Ingeniería Eléctrica
Universidad de la República, Montevideo, Uruguay
{abarte,alicia,agomez}@iie.edu.uy

Abstract. We study and evaluate an automatic fingerprint classification algorithm that we apply over the fully manual identification system being used by the Dirección Nacional de Identificación Civil (DNIC). To be compatible with the existing system and provide a gradual transition into a fully automatic procedure we mimic the classification scheme being used by DNIC technicians, which is based on a four-class Vucetich system. The classification algorithm we use is based on the method by Karu and Jain [4]. Some modifications to the original algorithm are proposed and evaluated over images extracted from a 4 million fingerprint card archive maintained by DNIC. The algorithm was also tested on fingerprints from the same individuals taken at two different points in time (separated several years) to further evaluate its performance and consistency.

1 Introduction

This work is motivated by an ongoing collaboration between the Universidad de la República and the Dirección Nacional de Identificación Civil (DNIC) concerned with civil identification affairs in Uruguay. The goal of this joint project was to evaluate an automatic fingerprint classification system compatible with the fully manual method that has been used by DNIC for several years. DNIC's classification scheme is based on the Vucetich system, that has four fundamental classes: arch, right and left loop, and whorl. To maintain backward compatibility with the existing system and provide a gradual transition into a fully automated system, we have used the same classification scheme. To classify we use the algorithm by Karu and Jain [4] with some minor modifications. This algorithm is based on a singularities approach, where heuristic criteria based on the number and position of singularities are used to classify among the different classes. The main modification we introduce is the addition of a masking step, that prevents having spurious singularities (core or delta points) in the background portion of the image. These are commonly caused by the presence of written words, lines, or noisy image areas. After successive and progressive tuning steps, the algorithm showed an acceptable overall performance. To evaluate the algorithm we extracted a representative sample of more than 4 hundred individual fingerprint

cards from the national archive held by DNIC. Each card has a ten-print image and the corresponding manual classification formula provided by human experts. The sampling process to obtain this database was designed to closely represent the natural distribution of patterns in the full 4 million population. As additional validation we test the algorithm with fingerprints of the same individuals but from different points in time separated several years.

The work is organized as follows: in Section 2 we review the Vucetich classification system used by DNIC, and mimicked by our system. In Section 3 we describe how we obtained the sample database, we give resulting image characteristics, and present percentage distributions of the different classes within the database. In Section 4 we review the main stages of the classification algorithm by Karu and Jain, and describe the main modifications we have introduced. In Section 5 we present the results and performance of the algorithm tested on our database. Finally, in Section 6 we outline the conclusions of this work.

2 The Vucetich Classification Scheme

Based on a forty-class scheme first proposed by Galton, Juan Vucetich introduced in 1896 a four-class fingerprint classification scheme [2]. This scheme has been in use for several years at DNIC offices and is the one we use in this work. The four fundamental classes are: arch, left loop, right loop and whorl. Arches have no singular points, loops present one core-delta pair, and whorls two core-delta pairs. This is similar to Henry's classification scheme, except that tended arches are taken as arches and twin loops as whorls. Here we are only interested in the coarse level classification of fingerprints. By coarse classification we mean the identification of fingerprints patterns in the four fundamental classes. Usually we need to further subdivide each class into several sub-classes. This process is being called sub-classification and is contemplated in the Vucetich classification scheme. As sub-classification is of singular importance for the effective retrieval of images from the database, the work we have done on automatic ridge counting between singularities will be reported elsewhere. Sometimes it is not easy to classify certain images, even for human experts. This can be due to bad prints, damaged or injured fingers, or simply caused by questionable patterns. In the latter case differences in the degree of judgment and interpretation of the individual classifying fingerprints may result in different classes. In the Vucetich system these are called transition figures. Regardless of the specific classification, the automatic algorithm should guarantee a consistent decision, not always achievable in manual classification systems.

3 Fingerprint Database

DNIC holds the fingerprints of over four million people. The fingerprints of each individual are stored in a paper card as shown in Fig. 1. The card archive is indexed and physically ordered by the ten fingerprint classification formula of each individual. In order to test the classification algorithm, the hole paper card

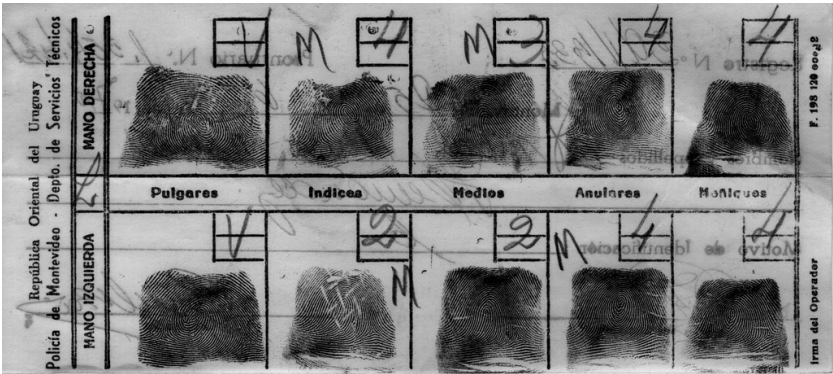


Fig. 1. Sample fingerprint card from the DNIC archive.

archive was sampled and more than four hundred cards were digitized to obtain over 4000 fingerprint digital images. The cards were chosen at random but evenly distributed over the hole archive to maintain the natural distribution of figures. No quality requirement was imposed on the selected cards. The cards were digitized at a resolution of 500 dpi and the fingerprints cut to become 512×480 pixels images. For each digitized card, all the information available (classification and sub-classification given by DNIC experts) was gathered. In Tables 1 and 2 we show some statistics on the sample fingerprint database. Observe that the distribution of figures is similar to that of the standard databases of NIST 3.

Table 1. Distribution for the right thumb finger in the four classes of the Vucetich scheme. We show the individual frequencies and the corresponding percentage values.

Class	Frequency	Percentage
Arch	22	5.07
Left loop	2	0.46
Right loop	203	46.77
Whorl	207	47.70
	434	

4 Classification Algorithm

It is well known the existence of efficient and robust fingerprint classification algorithms, specially the PCASYS approach by NIST 11. For evaluation purposes we were mainly interested in having a simple algorithm with ease of implementation that should provide reasonable results. For this we choose the algorithm by Karu and Jain 4 for a four class problem. We now describe its main stages

Table 2. Distribution for all the fingers in the four classes of the Vucetich scheme. We show the individual frequencies and the corresponding percentage values.

Class	Frequency	Percentage
Arch	459	10.56
Left loop	1281	29.47
Right loop	1282	29.49
Whorl	1325	30.49
	4347	

and the minor modifications we have introduced. The first thing is to compute a directional image, corresponding to ridge directions at each pixel in the input image. All possible directions are quantized to eight discrete values, equally spaced around the unit circle. Directions are then converted to vector form (*i.e.* (x,y) coordinate pairs), giving an appropriate representation to perform smoothing operations. The original image is averaged in 8×8 windows giving a reduced directional image of size 64×64 . Singularities are located computing the Poincaré index at every pixel (in the reduced image) in a 2×2 neighborhood. Doing this we label each pixel as either: ordinary, delta, or core point. Based on the number and location of delta and core points, the fingerprint is adequately classified. As unwanted spurious singularities may appear, an iterative approach is taken where the directional image is successively smoothed, until we can classify it. If after several smoothing steps we still can not classify the fingerprint, then it is tagged as unknown. As any classification approach based on singularities, the main drawback of the algorithm is the influence of spurious singularities that may cause classification errors. Unwanted singular points may originate from many sources: low quality image within the fingerprint area, spurious artifacts outside the fingerprint area: typed words, lines, etc. In [2] to alleviate this problem, vectors in the reduced directional image were divided by their distance to the center of the image, thus eliminating noise in the border areas. Instead, we take a different approach and choose to apply a masking step on the directional image that allows ignoring pixels in the background image area. Although similar to the one used in PCASYS [1], this one is much more simple but still greatly reduces classification errors. The mask is constructed in three steps: apply an automatic threshold, perform some geometry regularization, and include border image areas within the mask. As previously mentioned, the goal of the masking step is to extract the area within the image that corresponds to the fingerprint itself, ignoring the background pixels. Assuming a bimodal histogram (background and fingerprint image areas), the problem is to determine the best threshold separating the two modes. For this we use an automatic algorithm described in [3] pp. 20. This approach assumes that gray value observations come from a mixture of two Gaussians distributions having respective means and variances. Minimizing an adequate measure¹ results in the optimum threshold value

¹ This measure is being called the Kullback directed divergence, see [3] for details.

separating both modes present in the image. After applying the thresholding step we end up with a binary image highly irregular. Over this image we apply some morphology operations to regularize the mask. Specifically, some dilatation is first applied to fill the spaces between ridges and eliminate holes, and we then erode the result to better approximate the actual fingerprint area. In the third stage, since we are interested in locating cores and deltas in the center of the image, following [1], we arbitrarily fix a 30 pixel wide strip along the borders as part of the mask. Finally, to apply the mask we set the corresponding pixel in the directional image to zero. See Fig. 2.

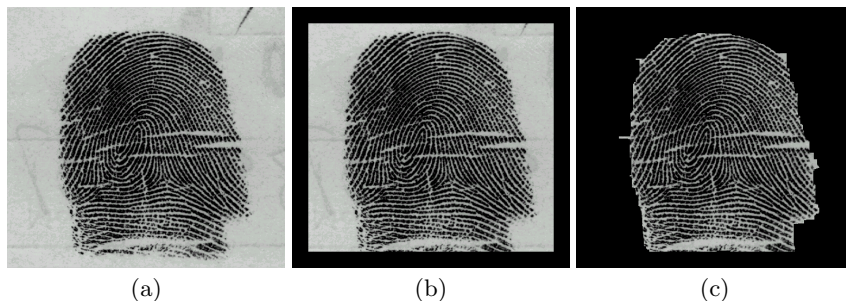


Fig. 2. Fingerprint image showing the effect of the masking operation; (a) original image; (b) 30 pixel wide strip along the borders; (c) mask over the image. Singularities located in the black area are not considered in the classification process.

5 Results and Evaluation

We test the classification algorithm on our image database. An iterative adjusting procedure was adopted in order to reduce the classification error percentage. Removal of spurious core-delta pairs, mask regularization levels, and valid positions of core points were conveniently selected. Normally each fingerprint was assigned one of the four fundamental classes. If after several smoothing steps we still can not find a classification, we tagged it as unknown. We present in Table 3 the resulting confusion matrix. To analyze the overall performance we divide fingerprints in four categories: correctly classified, wrongly classified, low quality images, and questionable or transition patterns. Correctly and wrongly classified means whether or not our classification coincides with the one provided by DNIC technicians. As the acquisition was done without quality control we have many low quality impressions, where the classification task is very difficult even for human experts. Criteria to decide whether an image was acceptable were jointly discussed with DNIC experts. Accordingly, the whole database was traversed to identify these fingerprints. The situation is similar with questionable or transition patterns. In Table 4 we show the overall percentage values for each category. Although the overall error percentage is certainly low (less than 10%),

Table 3. Classification results on our fingerprint database.

Assigned class	DNIC class			
	Arch	Left loop	Right loop	Whorl
Arch	71	20	17	7
Left loop	39	789	7	38
Right loop	22	5	748	52
Whorl	15	81	81	866
Unknown	3	23	20	46
	150	918	873	1009

Table 4. Overall database percentage distribution.

	Quantity Percentage	
Correct	2474	83.9
Errors	186	6.3
Low quality	227	7.7
Questionable	63	2.1
	2950	

if we consider the ten-print classification formula the error percentage drops drastically. That is, since errors from single fingerprints classification accumulate, the probability to have an error in the ten digits is very high. If we look at the location of singularities detected by the algorithm and compare them with the ones given by human experts, we find that subtle differences exist between both locations. Although the classification is still correct, this causes deviations in ridge counting based sub-classification schemes. The problem is a result of the smoothing operation performed within the classification loop and the difference in location is more noticeable in images that undergo several smoothing steps.

5.1 Consistency

As a cross validation test we probe the algorithm with fingerprints of the same individuals but from different points in time. Of course, as fingerprints remain the same with age, we expect to have the same classification formula for each set of images. We take three different impressions for the ten fingers in five individuals. Two recent simultaneous prints and one extracted from the DNIC archive (from ten years ago). For the two simultaneous prints low error percentages were expected, since the only difference is on the acquisition process being held by technicians. As predicted, the algorithm behaves very good obtaining almost no errors: 96% (1 out of 50). Comparing the first two sets with the third one, although the overall success percentage is still high (78%), we obtain some errors mainly motivated by poor quality impressions. Many of these errors can be eliminated by first applying a rejection criteria based on an appropriate image quality measure.

6 Concluding Remarks

In this work we study and evaluate an automatic fingerprint classification algorithm that we apply over the fully manual identification system being used by DNIC. The primary goal was to make a gradual transition into a fully automatic system. For this we intend to mimic the classification scheme being used by DNIC technicians, which is based on a four-class Vucetich system. The classification algorithm we use is based on the method by Karu and Jain [2]. Some modifications to the original algorithm are proposed and evaluated over images extracted from a 4 million fingerprint card archive been held by DNIC. Although some improvements can be made to the classification algorithm, preliminary results are very satisfactory specially considering the simplicity of the algorithm. Given the heterogeneity of fingerprints obtained from the database, it becomes mandatory the need of a *quality measure* to reject bad impressions. As in the original algorithm [4] we try rejecting images based on two different criteria: we set an upper limit on the amount of smoothing that can be applied to the directional image, and we try a global quality measure based on an average of the local homogeneity of the directional field (excluding the singularities) over the whole mask. The first one proved to be very effective, whereas the second one gives uneven results and does not seem to be an effective measure to reject bad images. The design of an effective quality measure is intended for future stages of the project.

Acknowledgments

We would like to acknowledge A. Spagenberg and S. Nesmachnow who also participated in the joint project between the Universidad de la República and the DNIC, Insp. Ppal. (PA) María del Carmen Almada Perotti, Crio. Raúl Marcora, DNIC staff, and A. Almansa for helpful discussions.

References

1. G. T. Candela et al., "PCASYS - A Pattern-Level Classification Automation System for Fingerprints". NIST Technical Report NISTIR 5163, Aug. 1995.
2. Dirección Nacional de Identificación Civil, "Manual de Dactiloscopia", Ministerio del Interior 1993.
3. Robert M. Haralick and Linda G. Shapiro, Computer and Robot Vision Volume I. Addison-Wesley, 1992.
4. K. Karu and A. Jain, "Fingerprint classification", *Pattern Recognition*, Vol. 29, No. 3, pp. 389-404, 1996.
5. C. I. Watson, "NIST Special Database 9, Mated Fingerprint Card Pairs". National Institute of Standard and Technology (February 1993).

Quality Measures of Fingerprint Images

LinLin Shen, Alex Kot, and WaiMun Koo

School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore, 639798
P150176565@ntu.edu.sg

Abstract. In an automatic fingerprint identification system, it is desirable to estimate the image quality of the fingerprint image before it is processed for feature extraction. This helps in deciding on the type of image enhancements that are needed and in deciding on thresholds for the matcher in the case that dynamic thresholds are used. In this paper, we propose a Gabor-feature based method for determining the quality of the fingerprint images. An image is divided into $N \times W \times W$ blocks. Gabor features of each block are computed first, then the standard deviation of the m Gabor features is used to determine the quality of this block. The results are compared with an existing model of quality estimation. Our analysis shows that our method can estimate the image quality accurately.

1 Introduction

Fingerprints have been widely accepted as one form of personal identification in criminal investigation, access control, and Internet authentication due to its unchangeability and uniqueness. Most available systems use the minutiae matching for identification. However, the minutiae-based approach is very sensitive to noise and deformation. For instance, false ridge endings and bifurcations may appear due to blurred or overinked problem. Moreover, ridge endings and bifurcations may disappear when a finger is pressed too hard or too light [1]. In other words, the performance of minutiae extraction algorithms relies heavily on the quality of the fingerprint. If we can examine the quality of the image first, we can reject the image with very poor quality. We can also combine the useful information into the procedure of enhancement, post-processing and matching to improve the identification process. It is therefore desirable to design an automatic scheme that examines and quantifies the quality of an acquired fingerprint image before it is processed.

Ratha and Bolle [9] proposed a method for image quality estimation in the wavelet domain, which is suitable for WSQ compressed fingerprint images. But it is not a desirable approach for uncompressed fingerprint image database since the wavelet transform consumes much computation. We still need to develop techniques in the spatial domain for accurate estimation of the fingerprint image quality. Hong and et.al. [2] proposed a method to quantify the quality of a fingerprint image by measuring the variance of gray levels, which is computed in a direction orthogonal to

the orientation field in each block. The variance is then used to decide the quality of the fingerprint image in terms of the image contrast of the block under consideration. However, this method is based on the orientation field and it doesn't take the ridge frequency into account. If the image is very noisy, the precise orientation field can't be computed. In addition, the computation is very complex.

Since Gabor feature can represent the ridge structures of fingerprints, in this paper, we propose an elegant method for computing image quality by Gabor features. Due to the accurate estimation, the results can be further used for foreground/background segmentation and smudginess and dryness computation. We also compare our algorithm with the variance method [2] described above by experiments. Our experimental results show that our method can estimate the image quality accurately.

2 Gabor Features

The Gabor-filter-based features, directly extracted from gray-level fingerprint images, have been successfully and widely applied to texture segmentation [5,6], face recognition, and handwritten numerals recognition [7]. In fingerprint applications, the Gabor-filter-based features for enhancement, classification and recognition are also proposed in [4]. The characteristics of the Gabor filter, especially for frequency and orientation representations, are similar to those of the human visual system.

In [7], the general form of a 2D Gabor filter is defined by

$$h(x, y, \theta_k, f, \sigma_x, \sigma_y) = \exp\left[-\frac{1}{2}\left(\frac{x_{\theta_k}^2}{\sigma_x^2} + \frac{y_{\theta_k}^2}{\sigma_y^2}\right)\right] \times \exp(i2\pi f x_{\theta_k}), \quad (1)$$

$$k = 1, \dots, m.$$

where $x_{\theta_k} = x \cos \theta_k + y \sin \theta_k$ and $y_{\theta_k} = -x \sin \theta_k + y \cos \theta_k$, f is the frequency of the sinusoidal plane wave, m denotes the number of orientations, θ_k is the k th orientation of the Gabor filter, and σ_x and σ_y are the standard deviations of the Gaussian envelope along the x and y axes, respectively.

Since most local ridge structures of fingerprints come with well-defined local frequency and orientation, f can be set by the reciprocal of the average inter-ridge distance and the value of θ_k is given by $\theta_k = \pi(k-1)/m$, $k = 1, \dots, m$. After deciding the parameters of the Gabor filter, the magnitude Gabor feature at the sampling point (X, Y) can be defined as follows:

$$g(X, Y, \theta_k, f, \sigma_x, \sigma_y) = \left| \sum_{x=-w/2}^{w/2-1} \sum_{y=-w/2}^{w/2-1} I(X+x, Y+y) h(x, y, \theta_k, f, \sigma_x, \sigma_y) \right|, \quad (2)$$

$$k = 1, \dots, m.$$

where $I(.,.)$ denotes the gray-level value of the pixel $(.,.)$, w is the size of the blocks (the image is divided into $N \times w \times w$ blocks). Once the parameters of the Gabor filter are determined, $m \times w \times w$ Gabor matrices are obtained. Then, each block is sampled by these matrices and m Gabor features are obtained. A $w \times w$ block is then compressed to m meaningful Gabor features.

3 The Proposed Method

After obtaining m Gabor features, g_{θ_k} , of the block, the standard deviation value G is computed as follows:

$$G = \left(\frac{1}{m-1} \sum_{k=1}^m (g_{\theta_k} - \overline{g_{\theta}})^2 \right)^{\frac{1}{2}}, \quad \overline{g_{\theta}} = \frac{1}{m} \sum_{k=1}^m g_{\theta_k} \quad (3)$$

where $\theta_k = \pi(k-1)/m$, $k = 1, \dots, m$. We observe that: (i) for good quality image blocks with local ridge orientation, the values of one or several Gabor features are much bigger than the value of others. (ii) for poor quality image blocks or background blocks without local ridge orientation, the values of m Gabor features are close to each other. So, the standard deviation value of the m Gabor features, G , is used for both foreground/background segmentation and image quality estimation.

A fingerprint image usually consists of a region of interest (ridges and valleys) along with a background (see Fig 2). We need to segment the fingerprint area (foreground) to avoid extraction of features in the noisy background areas of the image. We compute the value of G for each block. If G is less than a threshold value T_b , the block is marked as a background block, otherwise the block is marked as a foreground block. The quality field for the fingerprint image in Figure 2(a) is shown in Figure 2(b). The segmented image is shown in Figure 2(c).

The quality field value for a foreground block is defined to have one of the following values: good and poor. A block is marked as a poor quality block if its G value is less than a preset threshold T_q , otherwise it is marked as a good quality block. QI (Quality Index) is defined to quantify the quality of a fingerprint image, where

$$QI = 1 - \frac{\text{Number of "poor" Foreground Blocks}}{\text{Number of Foreground Blocks}} \quad (4)$$

A fingerprint image is marked as a good quality image if the QI value is bigger than a threshold T_Q , otherwise it is marked as a poor quality image. The choice of T_q and T_Q are determined experimentally.

When a fingerprint image is marked as a poor quality image, SI (Smudginess Index) and DI (Dryness Index) are used to determine whether this image consists of large number of dry blocks or smudged blocks. The idea is that for a smudged block, most ridges are connected with each other, so that the mean value of the block is small. While for a dry block, some of the ridges are disjointed and the mean value of the block will be larger. A poor block is marked as a smudged block if its mean value is less than a preset threshold T_s , while a poor block is marked as a dry block if its mean value is larger than another preset threshold T_d . Both T_s and T_d are determined by the mean value of the foreground blocks of the image.

$$SI = \frac{\text{Number of "poor" \& "smudged" Foreground Blocks}}{\text{Number of Foreground Blocks}} \quad (5)$$

$$DI = \frac{\text{Number of "poor" \& "dry" Foreground Blocks}}{\text{Number of Foreground Blocks}} \quad (6)$$

Two Thresholds T_s and T_d are chosen empirically to determine the type of a poor quality fingerprint image. If $SI \geq T_s$ and $DI \geq T_d$, the image is marked as others. If $SI \geq T_s$ and $DI < T_d$, the image is marked as smudged. If $SI < T_s$ and $DI \geq T_d$, the image is marked as dry.

In summary, our quality estimation algorithm can be stated as follows:

1. Divide the image I into N blocks of size $w \times w$.
2. Compute the m $w \times w$ Gabor matrices using eqn. (1).
3. For each block centered at pixel (i,j) , compute m Gabor features and G by eqns. (2) and (3) respectively.
4. Use the value of G to segment foreground/background of the image.
5. Compute the value of QI by eqn. (4), determine the quality type (good or poor) of the image.
6. If the image is a poor image, use SI and DI to determine the type (dry or smudged) of the image.

4 Experimental Results

The fingerprint database used in the experiment consists of 540 fingerprint images from 108 different fingers with 5 fingerprint images for each finger. The size of the fingerprint images is 300 x 300 pixels with resolution of 300dpi. The fingerprint images are captured using a capacitive fingerprint sensor and quantified into 256 gray levels. The images in the database are divided into 4 classes by visual inspection of their image qualities. 453 images are marked as good images, 39 are poor images from wet fingers (marked as smudged), 25 are poor images from dry fingers

(marked as dry) and 23 are poor images caused by other reasons (marked as others). So the percentage of poor images is 16%. To test the algorithm, each image is divided into 20×20 blocks ($N=225$, $w=20$). Table 1. shows the QI , SI and DI of some typical fingerprint images like Figure 1(a) (wet), Figure 1(b) (dry) and Figure 2(a) (normal).

After computing QI , SI and DI of a given fingerprint image, we use these 3 values to determine the type of the image as described early. We also ran the variance algorithm described in [2]. The results are shown in Table 2. To design the parameter of Gabor filters, we set $f = 0.12$, $\sigma_x = 4.0$, $\sigma_y = 4.0$, which are determined empirically. We also set $m = 8$, $\theta_k = \pi(k - 1) / m$, $k = 1, \dots, m$.

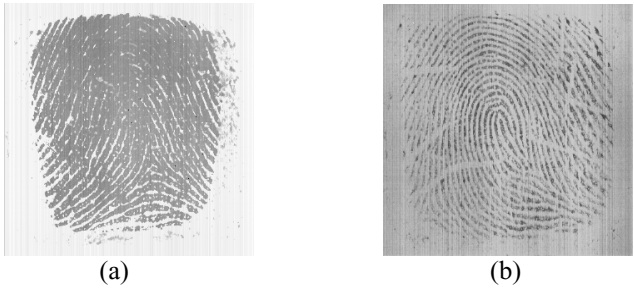


Fig. 1. Typical fingerprint images: (a) Wet Finger; (b) Dry finger;

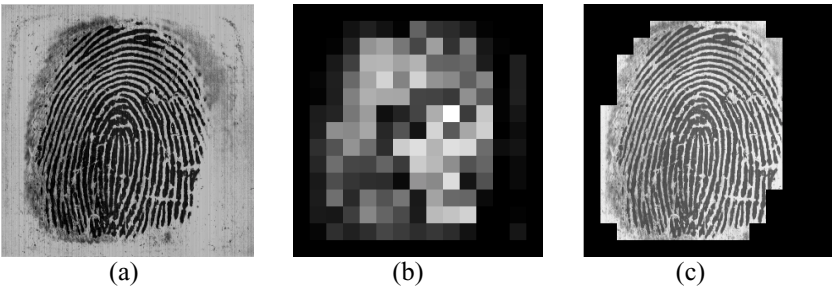


Fig. 2. Foreground/background segmentation: (a) origin image; (b) quality field (Standard deviation of m Gabor features); (c) segmented image.

Table 1. QI , SI , and DI of typical fingerprint images.

	Fig 1. (a)	Fig 1. (b)	Fig 2.(a)
QI	0.2314	0.2951	0.5714
SI	0.5950	0.2831	0.1494
DI	0.0661	0.4216	0.1299

Table 2. Results by our proposed Algorithm and (Variance Algorithm [2]).

Assigned Classes	True Classes			
	Good	Smudged	Dry	Others
Good	428 (423)	3 (6)	3 (5)	0 (1)
Smudged	18 (17)	36 (33)	0 (2)	0 (0)
Dry	6 (6)	0 (0)	22 (18)	0 (1)
Others	1 (7)	0 (0)	0 (0)	23 (21)
Accuracy	94.4% (93.3%)	92.3% (84.6%)	88% (72%)	100% (91.3%)

5 Discussions

In this paper, we proposed a method to determine the quality of a fingerprint image with Gabor feature. From Table 2, we observe that when good quality image measure is considered, the performance of variance algorithm (accuracy: 93.3%) is comparable with our algorithm (accuracy: 94.4%). However, when used to classify the poor quality image, our algorithm outperforms the variance algorithm. The reason is that in such situation, we can't get the accurate orientation field because of the poor quality of the image. In addition, the variance algorithm needs to compute the orientation field, which makes it much more complex in computation than our algorithm. Our approach has the advantage of fastness in computation, which is an important factor for fingerprint identification problem.

References

1. A.K. Jain, L. Hong, and R. Bolle, On-line fingerprint verification, *IEEE Trans. Pattern Analysis Machine Intelligent*, Vol. 19, No. 4, pp. 302-314, 1997.
2. Lin Hong, Yifei Wan, and Anil Jain, Fingerprint Image Enhancement: Algorithm and Performance Evaluation, *IEEE Transactions on PAMI*, Vol. 20, No. 8, pp. 777-789, August 1998.
3. Nalini K. Ratha, Shaoyun Chen, and Anil K. Jain. Adaptive flow orientation based feature extraction in fingerprint images, *Pattern Recognition*, Vol.28, pp. 1657-1672, 1995.
4. Chih-Jen Lee and Sheng-De Wang. A Gabor filter-based approach to fingerprint recognition, *1999 IEEE Workshop on Signal Processing Systems, SiPS 99*. pp. 371-378, 1999.
5. T.P. Weldon, W.E. Higgins, and D.F. Dunn, Efficient Gabor filter design for texture segmentation, *Pattern Recognition*, Vol.29, No.12, pp. 2005-2015, 1996.
6. A.K. Jain and F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, *Pattern Recognition*, Vol.24, No.12, pp. 1167-1186, 1991.
7. Y. Hamamoto, S. Uchimura, M. Watanabe, etc. A Gabor filter-based method for recognizing handwritten numerals, *Pattern Recognition*, Vol.31, No.4, pp. 395-400, 1998.
8. R. Cappelli, A. Erol, D. Maio, and D. Maltoni. Synthetic Fingerprint-image Generation, *Proceedings of International Conference on Pattern Recognition (ICPR2000)*, Barcelona, September 2000.
9. Nalini K. Ratha and R. Bolle. Fingerprint Image Quality Estimation, pp. 819-823, ACCV 2000.

Automatic Gait Recognition by Symmetry Analysis

James B. Hayfron-Acquah, Mark S. Nixon, and John N. Carter

University of Southampton, Southampton S017 1BJ, United Kingdom

{jbha99r,msn,jnc}@ecs.soton.ac.uk

Abstract. We describe a new method for automatic gait recognition based on analysing the symmetry of human motion, by using the Generalised Symmetry Operator. This operator, rather than relying on the borders of a shape or on general appearance, locates features by their symmetrical properties. This approach is reinforced by the psychologists' view that gait is a symmetrical pattern of motion and by other works. We applied our new method to two different databases and derived gait signatures for silhouettes and optical flow. The results show that the symmetry properties of individuals' gait appear to be unique and can indeed be used for recognition. We have so far achieved promising recognition rates of over 95%. Performance analysis also suggests that symmetry enjoys practical advantages such as relative immunity to noise and missing frames, and with capability to handle occlusion.

1 Introduction

Recently, there has emerged a new application domain of computer vision dealing with the analysis of human images. This includes ear and face recognition, body tracking and hand gesture recognition, to mention just a few. Recently, gait recognition has been added to this domain. As a biometric, gait concerns recognising people by the way they walk. One major advantage of gait over other biometrics (e.g. fingerprints) is that it does not require contact. Further gait is difficult to disguise or conceal, in application scenarios like bank robbery. Currently, gait is also the only biometric at a distance. Though it could be argued that physical condition factors such as drunkenness, pregnancy and injuries involving joints can affect an individual's motion, these factors are similar in principle to factors affecting other biometrics. The aim of gait recognition is to recognise people regardless of the clothes worn or the differing background. There have been allied studies of gait, notably among these are medical studies, psychological studies, modelling human motion and tracking people. Amongst these, psychologists suggest gait is a symmetrical pattern of motion[2] and that humans perceive gait as unique.

Although gait recognition is a fairly new research area, there is already a number of approaches. In the spatio-temporal approach, which is probably the earliest, the gait signature was derived from the spatio-temporal patterns of a walking person[7]. The different patterns of the motions of the head and the legs in translation and time were extracted. The patterns were then processed to determine the motion of the bounding contours from which a five-stick model was fitted. The gait signature was then

derived by normalising the fitted model in terms of velocity, that is by linear interpolation, and encouraging (85%) recognition rates were achieved.

In [4], optical flow was used to derive the gait signature by analysing the motion content (shape of motion) of a human walking. Generic object-motion characterisation is also another approach where the gait signature is derived from a parametric eigenspace[5] and the approach was applied to a database of seven subjects with ten image sequences each. The recognition rates were 88% and 100% for 8 and 16 eigenvectors, respectively. The approach was extended[3] to use canonical analysis, a model free approach to reduce the dimensionality of the input data whilst optimising class separability. Recently, Shutler et al extended statistical gait recognition via temporal moments [10]. This derived statistics with an intimate relationship to gait, with symmetry properties. In [6], gait signatures were derived from the frequency components of the variations in the inclination of the human thigh. As pendula modelled the periodic motion of the thigh during walking, this again suggests that symmetry analysis is suited to gait recognition.

2 Symmetry and Its Extraction

Symmetry is a fundamental (geometric) property suggesting it to be an important principle of perception[9]. An object is said to be symmetric if its shape is invariant to the application of symmetry operations. Boolean symmetry operations can only assess symmetry when the shape of the object is known in advance, rendering them inefficient. The discrete symmetry operator can estimate symmetry without the knowledge of the object's shape, unlike feature extraction operators that find a shape by relying on its border. The symmetry transform assigns a symmetry measure to each point in the image and is determined with respect to a given point-symmetry group. It has also been shown that the performance of the symmetry transform is not affected by existence of several objects in the scene[9].

To extract the symmetry of a walking human subject, feature templates are extracted from gait sequences to give template sequences. The symmetry operator uses an edge map of images in the sequences to assign symmetry magnitude and orientation to image points, accumulated at the midpoint of each pair of points.

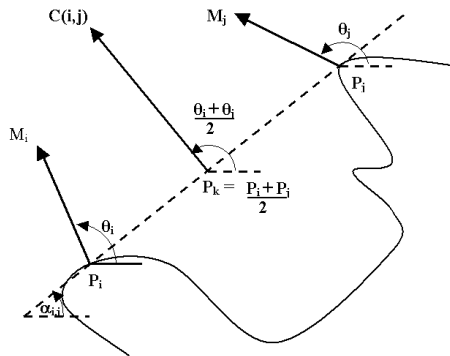


Figure 1. The symmetry contribution of edge points P_i and P_j .

The symmetry relation or contribution, $C(i,j)$ between the two points P_i and P_j is:

$$C(i,j) = D_{i,j} Ph_{i,j} I_i I_j \quad (1)$$

The symmetry distance weighting function, D , is defined as the minimum effort required to turn a given shape into its symmetric shape. It reflects the distance between two different points P_i and P_j , and is calculated as:

$$D_{i,j} = \frac{1}{\sqrt{2\pi\sigma}} \exp \left(- \left(\frac{\|P_i - P_j\| - \mu}{2\sigma} \right)^2 \right), \forall i \neq j \quad (2)$$

where σ controls the scope of the function. Each value of σ implies a different scale thus making it suited to multi-resolution schemes. A large value of σ implies large-scale symmetry that gives distant points similar weighting to close points. Alternatively, a small value of σ implies local operation and local symmetry. Recently a focus, μ , was therefore introduced into the distance weighting function to control the focusing capability of the function, hence further improving the scaling possibilities of the symmetry distance function. The addition of the focus into the distance weighting function moves the attention of the symmetry operator from points close together to a selected distance.

The logarithm intensity function, I_i , of the edge magnitude M at point (x,y) is $I_i = \log(1 + M_i)$. Using the logarithm of magnitude reduces the differences between high gradients or symmetries resulting from weak edges, making the correlation measure less sensitive to very strong edges. The phase weighting function between two points P_i and P_j is:

$$Ph_{i,j} = (1 - \cos(\theta_i + \theta_j - 2\alpha_{i,j}))(1 - \cos(\theta_i - \theta_j)), \quad \forall i \neq j, \quad \alpha(i,j) = \text{atan}\left(\frac{y_i - y_j}{x_i - x_j}\right) \quad (3)$$

is the angle between the line joining the two points and the horizon. The symmetry contribution value obtained is then plotted at the midpoint of the two points. The symmetry transform as discussed here detects reflectional symmetry. It is invariant under 2D rotation and translation transformations and under change in scale [9], and as such has potential advantage in automatic gait recognition.

3 Symmetry and Gait

3.1 Deriving the Gait Signature

To derive the gait signature for a subject an image sequence is used. The following gives an overview of the steps involved. First, the image background is subtracted from the original image, Fig. 2a to obtain the silhouette, Fig. 2b. The Sobel operator is then applied to the image in Fig. 2b to derive its edge-map, Fig. 2c. Where the gait signature is derived from optical flow information, the optical flow image is extracted from two successive silhouettes. The edge-map is thresholded so as to set all points beneath a chosen threshold to zero, to reduce noise or remove edges with weak strength, which may be due to the background removal. The symmetry operator is then applied to give the symmetry map, Fig. 2d. For each image sequence, the gait signature, GS , is obtained by averaging all the symmetry maps.

3.2 Gait Recognition

The Fourier transform was then applied to each of the gait signatures and the transform was low-pass filtered to reduce sensitivity to high-frequency components.

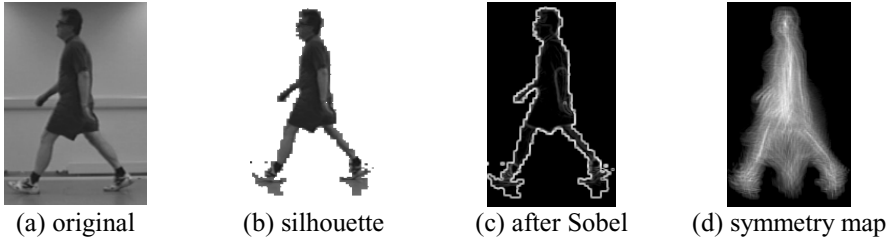


Figure 2. Images from the SOTON data.

Different cut-off frequencies were used to determine the appropriate number of Fourier components. For purposes of classification or recognition, the similarity differences between the Fourier descriptions of the gait signatures are then calculated using Euclidean distance. The magnitude spectra only were used here because they gave a better result than by using phase.

3.3 Recognition by Symmetry

The new method was applied to two different databases of spatial templates. The SOTON database has four subjects with four image sequences each and that of UCSD six subjects with seven image sequences of each. For both SOTON and UCSD databases, we derived gait signatures for silhouette and optical flow information. These provide alternative versions of the input data for our technique to process. The values for σ and μ used were 27 and 90, respectively, unless otherwise stated. The k -Nearest Neighbour rule was then applied for classification, using $k = 1$ and $k = 3$, as summarised in Table 1. The correct classification rates were 100% for both $k = 1$ and $k = 3$ for the SOTON database. For the UCSD database, the recognition rates for silhouette information were 97.6 and 92.9% for $k = 1$ and $k = 3$. A CCR of 92.9% was obtained for the optical flow information, for both $k = 1$ and $k = 3$.

Table 1: Initial results obtained from two disparate databases.

Database	# Subjects	# Sequences	Data Type	CCR (%)	
				$k = 1$	$k = 3$
SOTON	4	16	Silhouette	100	100
			Optical flow	100	100
UCSD	6	42	Silhouette	97.6	92.9
			Optical flow	92.9	92.9

For the low pass filter, all possible values of radius were used to investigate the number of components that can be used (covering 0.1 to 100% of the Fourier data).

Though the results of Table 1 were achieved for all radii greater than 3 (using the SOTON database), selecting fewer Fourier components might affect the recognition rates on a larger database of subjects, and this needs to be investigated in future.

3.4 Performance Analysis of Symmetry Operator

Performance was evaluated with respect to missing spatial data, missing frames and noise using the SOTON database. Out of the 16 image sequences in the database, one (from subject 4) was used as the test subject with the remainder for training.

Missing Frames: The evaluation, aimed to simulate time lapse, was done omitting a consecutive number of frames. For a range of percentages of omitted frames, Fig. 3a, the results showed no effect on the recognition rates for $k = 1$ or $k = 3$. This is due to the averaging associated with the symmetry operator. Fig. 3a shows the general trend of deviation of the best match of each subject to the test subject.

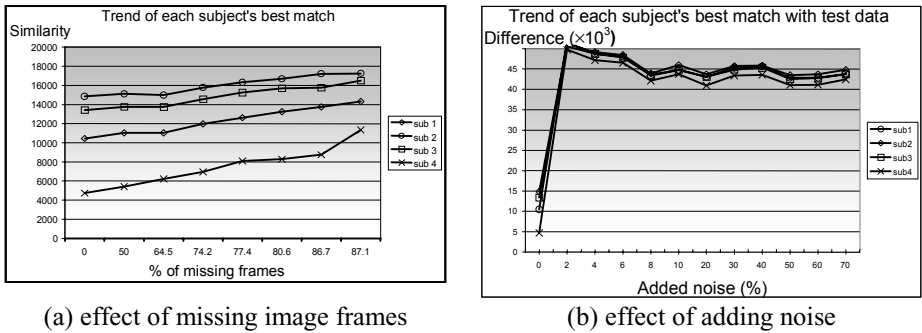


Figure 3. Performance Analysis: Omitted Frames and Addition of Noise.

Adding/Omitting Spatial Data: The evaluation was done by masking with a rectangular bar of different widths: 5, 10 and 15 pixels in each image frame of the test subject and at the same position. The area masked was on average 13.2%, 26.3% and 39.5% of the image silhouettes, respectively. The bar either had the same colour as the image silhouette or as the background colour, as shown in Fig. 4, simulating omission and addition of spatial data, respectively. In both cases, recognition rates of 100% were obtained for bar size of 5 pixels for both $k = 1$ and $k = 3$. For a bar width of 10 pixels, Fig. 4c failed but Fig. 4a gave the correct recognition for $k = 3$ but not for $k = 1$. For bar sizes of 15 and above, the test subject could not be recognised.

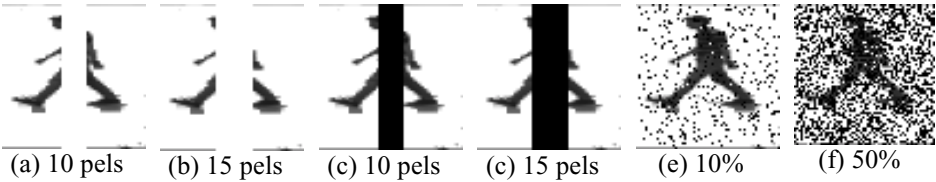


Figure 4. Occluded and Noisy Data.

Noise: To investigate the effects of noise, we added synthetic noise to each image frame of a test subject and compared the resulting signature with those of the other

subjects in the database. Fig. 4 shows samples of the noise levels used. The evaluation was carried out under two conditions. First by using the same values of σ and μ (eqn. 2) as earlier. For a noise level of 5%, the recognition rates for both $k = 1$ and $k = 3$ were 100%. For 10% added noise, the test subject could still be recognised correctly for $k = 1$ but not for $k = 3$. With added noise levels of 20% and above, the test subject could not be recognised for $k = 1$ or $k = 3$. With the second condition, the values of σ and μ were made relatively small. The recognition rates (100%) were not affected for both $k = 1$ and $k = 3$ for added noise levels even exceeding 60%. Fig. 3b shows how the best match of each subject deviated from the test subject as more noise was added.

4 Conclusions

The aim of this paper is to support the psychology view that the symmetry of human motion can be used for recognition. We have therefore presented, as a starting point, a new approach to automatic gait recognition. It has been shown that human gait appears to have distinct symmetrical properties that can be extracted for recognition. The symmetry operator, essentially, forms an accumulator of points, which are measures of the symmetry between image points to give a symmetry map. By using the symmetry operator, the Discrete Fourier Transform and a basic nearest-neighbour approach, the results have produced a recognition rate of 100% for both $k = 1$ and $k = 3$ on a small database of four subjects. Comparable recognition rates have been achieved using the same databases as in other works. The symmetry operator has been shown to handle missing spatial data, missing image frames, and to some extent noise. Thus, it will prove very useful when applied to poorly extracted sequences, partially occluded and missing frames in image sequences for gait recognition.

References

1. D. Cunado, M.S. Nixon, and J.N. Carter, Gait extraction and description by evidence gathering , *Proc. 2nd Int. Conf AVBPA99, Washington*, 1999, pp. 43-48.
2. J.T. Cutting, D.R. Proffitt, and L.T. Kozlowski, A biomechanical invariant for gait perception , *J. Exp. Psych.: Human Perception and Performance*, 1978, pp. 357-372.
3. P.S. Huang, C.J. Harris, and M.S. Nixon, Human gait recognition in canonical space using spatio-temporal templates , *IEE Proc. VISP*, April 1999, pp. 93-100.
4. J. Little and J. Boyd, Recognizing People by Their Gait: The Shape of Motion , *Videre*, **1**(2), 1-32, 1998.
5. H. Murase and R. Sakai, Moving object recognition in eigenspace representation: gait analysis and lip reading , *Patt. Recog. Lett.*, 1996, pp. 155-162.
6. M.S. Nixon, J.N. Carter, P.S. Huang, and S.V. Stevenage, Automatic Gait Recognition , In: *BIOMETRICS Personal identification in Networked Society*, Chapter 11, pp. 231-250. Kluwer Academic Publishers, 1999.
7. S.A. Niyogi and E.H. Adelson, Analysing and recognising walking figures in xyt , In: *Proc. CVPR*, 1994, pp. 469-474.
8. C.J. Parsons and M.S. Nixon, Introducing focus in the generalised symmetry operator , *IEEE Sig. Proc. Lett.*, **3**, 1999, pp. 49-51.
9. D. Reisfeld, H. Wolfson, and Y. Yeshurun, Context-free attentional operators: The generalised symmetry transform . *IJVC*, 1995, pp. 119-130.
10. J.D. Shutler, M.S. Nixon. and C.J. Harris, Statistical gait recognition via temporal moments . *4th IEEE Southwest Symp. on Image Analysis and Int.*, 2000, pp. 291-295.

Extended Model-Based Automatic Gait Recognition of Walking and Running

Chew-Yean Yam, Mark S. Nixon, and John N. Carter

Department of Electronics and Computer Science
University of Southampton, S017 1BJ Southampton, United Kingdom
{cyy99r,msn,jnc}@ecs.soton.ac.uk

Abstract. Gait is an emerging biometric. Current systems are either holistic or feature based and have been demonstrated to be able to recognise people by the way they walk. This paper describes a new system that extends the feature based approach to recognise people by the way they walk and run. A bilateral symmetric and coupled oscillator is the key concept that underlies this model, which includes both the upper and the lower leg. The gait signature is created from the phase-weighted magnitude of the lower order Fourier components of both the thigh and knee rotation. This technique has proved to be capable of recognising people when walking or running and future work intends to develop invariance attributes of walking or running for the new description.

1 Introduction

Using gait as a biometric is motivated by occlusion of criminals' faces and that they either walk or run to escape a crime scene. As such, even though many techniques have been developed to recognise people by the way they walk, there is no extant technique, which could recognise both by walking and by running. We describe a new system that can model both running and walking.

In literature, Aristotle and Leonardo da Vinci studied human movement and Shakespeare observed the possibility of recognition by gait. More recently, Murray^[1] produced standard walking movement patterns for pathologically normal men. Bianchi^[2] revealed that there are inter-individual differences in human gait mechanics. Since the action of walking is dictated by the skeleto-muscular structure and the same structure is applied to running, it suggests that if gait is indeed unique, then so should running. Perhaps, the earliest approach to gait recognition was to derive a gait signature from a spatio-temporal pattern^[3]. Images were projected into an eigenspace and the resulting eigenvectors were used for recognition^[4]. Then, the dense optical flow^[5] technique used the relative phases of optical flow to form a feature vector to create a signature. A more recent statistical based approach combined canonical space transformation based on canonical analysis with the eigenspace transformation. Later, temporal information obtained from optical-flow changes between two consecutive spatial templates was incorporated to improve recognition capability^[6]. The only model-based human gait recognition system^[7] models human walking as two interconnected pendula representing thigh motion,

which combined a velocity Hough transform with a Fourier representation to obtain a gait signature.

Walking may be described in terms of double support, where two limbs are in contact with the ground, and single support, where one foot is in contact with the ground. Running is the natural extension of walking, which involves increased velocities, different joint movement and coordination. The running cycle, however, is not distinguished from walking by velocity, but by whether a person becomes airborne during motion, with 2 periods of double float where neither foot is in contact with the ground. The way the foot contacts the ground is different for walking and for running. Li et. al. observed that there occur topological similarities in the co-ordination patterns between the thigh and lower leg in walking and running, which co-existed with functional differences throughout the gait cycle, especially in the transition from stance to swing phase., i.e. between 20% and 40% of the gait cycle^[8].

We describe a new gait model that can handle running and walking, and with fewer parameters, in section 2.1. We then show how this can be used to create a gait signature in section 2.2, that is shown to be able to recognise people by the way they walk and run in section 3, on a limited database.

2 Gait Modelling and Analysis

2.1 Coupled Oscillator Gait Model

As human gait is rhythmic and is naturally an oscillatory behaviour,^[9] we can assume that an oscillator controls each limb and that limb movement is interconnected or coupled in some way. The main characteristic of human gaits, including walking, running and sprinting is bilaterally symmetric where the left and right legs and opposite side of arms interchange with each other with a phase shift of half a period. Both legs perform the same motion but out of phase with each other by half a period. These motions operate in space and time, satisfying the rules of spatial symmetry (swapping legs) and temporal symmetry (a phase-lock of half a period in general). Fig. 1 shows the rotation both of thighs and knees for a walking subject. Hence, we can assume the legs are coupled oscillators with half a period of phase shift. Both legs can be modelled by two distinct but systematically coupled oscillators, which oscillate at the same frequency (frequency-lock) but with fixed relative phase difference.

The leg can be modelled as two pendula joined in series, see Fig. 2. The thigh rotation, $\theta_T(t)$, is described by Eq. (1), where t is the time index for the normalised gait cycle, A_T is the amplitude of the thigh rotation and C_T is the offset.

$$\theta_T(t) = A_T \cos(2\pi t) + C_T \quad (1)$$

Eq. (1) can be applied for both running and walking. Note that the gait cycles for running and walking are normalised so that they are invariant to speed. The knee rotation, $\theta_K(t)$, can be represented as

$$\theta_K(t) = \begin{cases} A_{K1} \sin^2(2\pi t) + C_{K1} & , 0 \leq t < p \\ A_{K2} \sin^2(2\pi(t + \phi)) + C_{K2} & , p \leq t < 1 \end{cases} \quad (2)$$

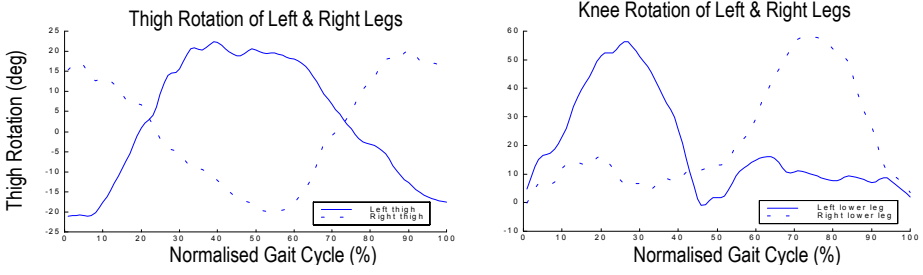


Fig. 1. (a) and (b) are the rotation of thighs and knees respectively, with half a period shift.

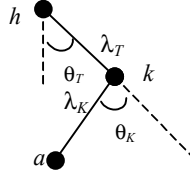


Fig. 2. The model of the thigh and lower leg: an upper pendulum models the thigh and the lower pendulum models the lower leg, connected at the knee joint.

where A_{K1} and A_{K2} are the amplitudes of the knee rotation, C_{K1} and C_{K2} are the offsets, ϕ is the phase shift and p is the time when the second double support starts (walking) and the double float starts (running). For walking, p is 0.4 whereas the thigh swings faster in running and then p is 0.3. Future work will aim to determine the effect of velocity on p . The \sin^2 term models well the basic motion as depicted in Fig. 1(b).

Given subject extraction, where the horizontal position is known or fixed, the horizontal displacement of the hip can be ignored. However, the vertical oscillation made during running is bigger than for walking, thus, the vertical displacement, $S_y(t)$, needs to be incorporated in the model, as given by

$$S_y(t) = A_y \sin(4\pi t) \quad (3)$$

where A_y is the amplitude of the vertical oscillation. Note that the frequency is twice the frequency of the leg motion, as a gait cycle comprises of two steps.

The structure of the thigh can be described by a point h that represents the hip and the line passing through h at an angle θ_T . The knee is then

$$k(t) = h(t) + \lambda_T u_T(t) \quad (4)$$

where $u_T(t)$ is the unit vector of the line direction, h is the position of the hip and λ_T is the thigh length, as $u_T(t) = [-\sin\theta_T(t), \cos\theta_T(t)]$ and $h(t) = [h_x(0), h_y(0) + S_y(t)]$, where $h_x(0)$ and $h_y(0)$ are the initial hip coordinates. Decomposing Eq. (4) into the x and y parts yields the coordinates of the knee point as,

$$k_x(t) = h_x(0) - \lambda_T \sin \theta_T(t) \quad (5)$$

$$k_y(t) = h_y(0) + S_y(t) + \lambda_T \cos \theta_T(t) \quad (6)$$

Similarly, the structure of the lower leg is given by a line which starts at the knee, that passes through k at an angle θ_k . The ankle a is

$$a(t) = k(t) + \lambda_K u_K(t) \quad (7)$$

where $u_K(t)$ is the unit vector of the line direction, $k(t)$ is the position of the knee and λ_K is lower leg length, as $u_K(t) = [-\sin(\theta_T(t) - \theta_K(t)), \cos(\theta_T(t) - \theta_K(t))]$ and $k(t) = [k_x, k_y]$, where k_x and k_y is the point of the knee. Decomposing Eq. (7) into x and y parts yields the coordinates of the ankle as,

$$a_x(t) = k_x(t) - \lambda_K \sin(\theta_T(t) - \theta_K(t)) \quad (8)$$

$$a_y(t) = k_y(t) + \lambda_K \cos(\theta_T(t) - \theta_K(t)) \quad (9)$$

2.2 Feature Extracti on, Fourier Description, and k -Nearest Neighbour

Equations (5, 6, 8 and 9), which describe the model of the moving leg, are used as the basis for feature extraction. The parameters of interest are $h_x(0)$, $h_y(0)$, A_T , A_{K1} , A_{K2} , A_y , C_T , C_{K1} , C_{K2} , λ_T , λ_K and ϕ . With this model, the computational cost and the number of parameters required is greatly reduced as compared with the earlier model-based approach^[8], which requires at least 26 parameters to approximate the motion of a single leg, not to mention both legs. With appropriate phase-lock, the model can handle both the left and right legs with the same number of parameters. By incorporating the coupled oscillators, the moving legs can be extracted accurately without confusion. Template matching is used frame by frame to determine the best values for the parameters and hence the angle of the line that best matches the edge data. These angles are then used to compute a Fourier transform representing the spectrum of variation in the thigh and lower leg.

The first and the second harmonic of the Fourier components of both thigh and knee rotations have the highest inter-class variance as compared with the higher components, which drop to near zero as the cut-off frequency of human walking is 5Hz. Multiplying the magnitude and phase component, i.e. phase-weighted magnitude, increases the inter-class variance. Hence, the first and second phase-weighted magnitude of both thigh and knee are used to create the gait signature. A basic classifier, the k -nearest neighbour was used. Clearly, other classifiers can be used to improve recognition capability but the issues here are rather more basic in nature. The Euclidean distance is used to obtain the distance between the test sample and the training data. The recognition results were evaluated by leave-one-out cross validation for different values of k in the k -nearest neighbour rule.

3 Results

The database consists of the side views of 5 subjects, each with 5 sequences of running and of walking on an electric treadmill with speeds set at 3 and 6mph. The subjects wore their own choice of clothing. The inclination of the thigh and knee, as extracted by the new automatic coupled oscillator technique, were compared with manually labelled angles of the grey level images. This analysis showed that the average difference of the automatically labelled data was 2.5° . This is reflected in Fig. 3, which shows the extracted position of the front of the leg superimposed on the original images. The extracted angles are still precise in regions where the legs cross and occlude each other and tolerates well with the nature of the background, especially occlusion by the treadmill's support.

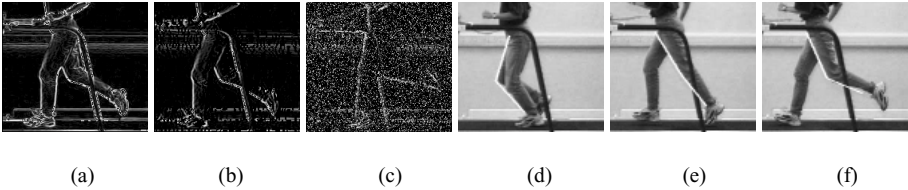


Fig. 3. (a) Edge data, (b) only leading edge is taken, (c) edge with 25% noise and (d-f) feature extraction results (superimposed in white).

Fig. 4 shows the phase-weighted magnitude of the Fourier component obtained from both the thigh and knee rotation. These are just three out of the four components used for recognition. The descriptors of walking subjects appear to cluster well, and are consistent with high recognition rates. The clusters for the running subjects are less distinct, though a more sophisticated classifier could handle such data, especially when complemented by other descriptors.

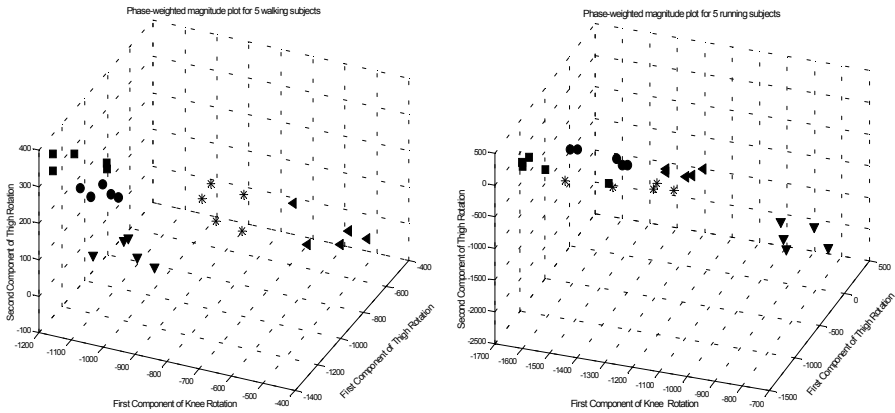


Fig. 4 (a) and (b) shows the phase-weighted magnitude obtained from the Fourier transform of 5 walking and running subjects, respectively.

The recognition rates are very encouraging and reach 96% for walking and 92% for running. These are consistent with other studies^[5,7] on similarly sized databases.

The effect of smoothing in the feature space by using larger values of k is only seen in the noisy running data. In other cases the nature of the feature space clusters lead to limited effect. On this data, it appears that $k=3$ is the most prudent choice in general. With 50% added noise, the rate maintains at an acceptable level, which reaches 80% and 76% for walking and running respectively. Table 1 shows the classification rates.

Table 1. The classification rates via k -nearest neighbour for walking and running, with noise level of 0%, 25% and 50%, with $k=1$, $k=3$ and $k=5$.

Noise Level (%)	Walking (%)			Runni ng (%)		
	$k=1$	$k=3$	$k=5$	$k=1$	$k=3$	$k=5$
0	96	96	96	92	88	84
25	80	88	88	84	84	88
50	80	80	68	64	72	76

4 Conclusions

A new model based technique has been developed for the thigh and lower leg and achieves fewer parameters by using the property of coupled oscillators. This new model has been shown to good effect in recognition of subjects by the way they walk and by the way they run, with a relatively better recognition rate for walking as compared to running. However, the recognition rate could be improved by using a more sophisticated classifier. Accordingly there is a potential for determining an invariance relationship between walking and running which in turn could be used to recognise people by either the way they walk or run.

References

1. M.P. Murray, A.B. Drought, and R. C. Kory. Walking Pattern of Normal Men. *J. Bone and Joint Surgery*, 46-A(2), 335-360, 1964.
2. L. Bianchi, D. Angeloni, and F. Lacquaniti, Individual characteristics of human walking mechanics, *Pfl gers Arch Eur J Physiol*, 436, 343 356, 1998.
3. S.A. Niyogi and E.H. Adelson. Analyzing and Recognizing Walking Figures in XYT. *Proc. IEEE Comp. Soc. Conf. on Comp. Vision and Patt. Recog.*, 469-474, 1994.
4. H. Murase and R. Sakai. Moving Object Recognition in Eigenspace Representation: Gait Analysis and Lip Reading. *Patt. Recog. Letters*, 17, 156-162, 1996.
5. J. Little and J. Boyd, Recognizing People by Their Gait: The Shape of Motion. *MIT Press Journal-Videre*, 1(2), 1-32, 1998.
6. P.S. Huang, C.J. Harris, and M.S. Nixon. Human gait recognition in canonical space using temporal templates. *IEE Proc.: VISP*, 146(2), 93-100, 1999.
7. D. Cunado, M.S. Nixon and J.N. Carter. Automatic Gait Recognition via Model-Based Evidence Gathering. *AutoID 99 Proc.: Workshop on Auto. Identification Adv. Tech.*, 27-30, October 1999.
8. L. Li, E.C.H. van den Bogert, G.E. Caldwell, R.E.A. van Emmerik, and J. Hamill, Coordination Patterns of Walking and Running at Similar Speed and Stride Frequency, *Human Mov. Sc.*, 18, 67-85, 1999.
9. I. Stewart. Symmetry-breaking cascades and the dynamics of morphogenesis and behaviour. *Sc. Progress*, 82(1), 9-48, 1999.

EigenGait: Motion-Based Recognition of People Using Image Self-Similarity

Chiraz BenAbdelkader¹, Ross Cutler², Harsh Nanda¹, and Larry Davis¹

¹ University of Maryland, College Park

{chiraz, rgc, nanda, lsd}@umiacs.umd.edu

² Microsoft Research, rcutler@microsoft.com

Abstract. We present a novel technique for motion-based recognition of individual gaits in monocular sequences. Recent work has suggested that the image self-similarity plot of a moving person/object is a projection of its planar dynamics. Hence we expect that these plots encode much information about gait motion patterns, and that they can serve as good discriminants between gaits of different people. We propose a method for gait recognition that uses similarity plots the same way that face images are used in eigenface-based face recognition techniques. Specifically, we first apply Principal Component Analysis (PCA) to a set of training similarity plots, mapping them to a lower dimensional space that contains less unwanted variation and offers better separability of the data. Recognition of a new gait is then done via standard pattern classification of its corresponding similarity plot within this simpler space. We use the k-nearest neighbor rule and the Euclidian distance. We test this method on a data set of 40 sequences of six different walking subjects, at 30 FPS each. We use the leave-one-out cross-validation technique to obtain an unbiased estimate of the recognition rate of 93%.

1 Introduction

Human gait has long been an active subject of study in biomechanics, kinesiology, psychophysics, and physical medicine [25, 21, 28]. The reasons and applications for this interest include: detection of gait pathologies, rehabilitation of an injured person, improving athletic performance, and designing ergonomic-based athletic and office equipment. These gait studies typically analyze 3D temporal trajectories of marked points on the body in terms of their frequency and phase. The relationships among the component patterns of the gait, such as phase differences between the trajectories, are also a valuable source of information.

In machine vision, gait recognition has received growing interest due to its emergent importance as a biometric [9, 4]. The term *gait recognition* is used to signify recognizing individuals by the way they walk in image sequences. Gait detection is the recognition of different types of human locomotion, such as running, limping, hopping, etc. Because human ambulation (gait) is one form of human movement, gait recognition is closely related to vision methods for detection, tracking and recognition of human movement in general (such as actions and gestures).

Gait recognition research has largely been motivated by Johansson's experiments [21] and the ability of humans for motion perception from Moving Light Displays

(MLDs). In these experiments, human subjects were able to recognize the type of movement of a person solely from observing the 2D motion pattern generated by light bulbs attached to the person. Similar experiments later showed some evidence that the identity of a familiar person ('a friend') [3], as well as the gender of the person [11] might be recognizable from MLDs, though in the latter case a recognition rate of 60% is hardly significantly better than chance (50%).

Despite the agreement that humans can perceive motion from MLDs, there is still no consensus on how humans interpret this MLD-type stimuli (i.e. how it is they use it to achieve motion recognition). Two main theories exist: The first maintains that people use motion information in the MLDs to recover the 3D structure of the moving object (person), and subsequently use the structure for recognition; and the second theory states that motion information is directly used to recognize a motion, without structure recovery. In machine vision, methods that subscribe to the former theory are known as *structure from motion* (SFM) [17], and those that favor the latter are known as motion-based recognition [8].

Consequently, there exist two main approaches for gait recognition each of which favors one of the two above theories. In SFM-based methods, a set of body points are tracked (as a result of body structure recovery), and their motion trajectories are used to characterize, and thereby recognize the motion or action performed by the body. Note that this approach emulates MLD-based motion perception in humans, since the body part trajectories are in fact identical to MLD-type stimuli. Furthermore, this approach is supported by biomedical gait research [28] which found that the dynamics of a certain number of body parts/points totally characterize gait. However, because tracking body parts in 3D over a long period of time remains a challenge in vision, the effectiveness of SFM-based methods remains limited.

Motion-based recognition methods, on the other hand, characterize the motion pattern of the body, without regard to its underlying structure. Two main approaches exist; one which represents human movement as a sequence (i.e. discrete number) of poses/configurations; and another which characterizes the spatiotemporal distribution generated by the motion in its continuum.

The method we describe in this paper takes a motion-based recognition approach. Our method makes the following assumptions:

- People walk with constant speed and direction for about 3-4 seconds.
- People walk approximately parallel to the image plane.
- The camera is sufficiently fast to capture dynamics of motion (we use 30Hz).

2 Related Work

We review vision methods used in detection, tracking and recognition of human movement in general, as they are closely related to gait recognition ([9, 11, 14] are good surveys on this topic). These methods can be divided into two main categories: methods that recover high-level structure of the body and use this structure for motion recognition, and those that directly model how the person moves. We shall describe the latter in more detail as it is more relevant to the gait recognition approach proposed in this paper.

2.1 Structural Methods

A 2D or 3D structural model of the human body is assumed, and body pose is recovered by extracting image features and mapping them to the structural components of the model (i.e. body labeling). Hence a human is detected in the image if there exists a labeling that fits the model well enough (based on some measure of goodness of fit) [17, 18, 32, 15, 53]. Once a person has been detected and tracked in several images, motion recognition is done based on the temporal trajectories of the body parts, typically by mapping them to some low-dimensional feature vector and then applying standard pattern classification techniques [2, 54, 4, 26].

2.2 Structure-Free Methods

To recognize a moving object (or person), these methods characterize its motion pattern, without regard to its underlying structure. They can be further divided into two main classes. The first class of methods consider the human action or gait to be comprised of a sequence of poses of the moving person, and recognize it by recognizing a sequence of static configurations of the body in each pose [27, 20, 16]. The second class of methods characterizes the spatiotemporal distribution generated by the motion in its continuum, and hence analyze the spatial and temporal dimensions simultaneously [29, 30, 12, 24, 23, 10]. Our method is closely related to the work of [10], in that both use similarity plots to characterize human motion, though we use them for gait recognition, and Cutler and Davis use them mainly for human detection.

State-Space Methods. These methods represent human movement as a sequence of static configurations. Each configuration is recognized by learning the appearance of the body (as a function of its color/texture, shape or motion flow) in the corresponding pose.

Murase and Sakai [27] describe a template matching method which uses the parametric eigenspace representation as applied in face recognition [55]. Specifically, they use PCA (Principal Component Analysis) to compute a 16-dimensional manifold for all the possible grey-scale images of a walking person. An input sequence of images (after normalization) is hence mapped to a trajectory in this 16-dimensional feature space, and gait recognition is achieved by computing the distance between the trajectories of the input image sequence and a reference sequence.

Huang et al. [20] use a similar technique, as they apply PCA to map the binary silhouette of the moving figure to a low dimensional feature space. The gait of an individual person is represented as a cluster (of silhouettes) in this space, and gait recognition is done by determining if all the input silhouettes belong to this cluster.

He and Debrunner [16] recognize individual gaits via an HMM that uses the quantized vector of Hu moments of a moving person's silhouette as input.

Spatiotemporal Methods. Here, the action or motion is characterized via the entire 3D spatiotemporal (XYT) data volume spanned by the moving person in the image. It could for example consist of the sequence of grey-scale images, optical flow images, or binary silhouettes of the person. This volume is hence treated as a 'large' vector,

and motion recognition is typically done by mapping this vector to a low-dimensional feature vector, and applying standard pattern classification technique in this space. The following methods describe different ways of doing this.

Of particular interest is the recent work by Cutler and Davis [10], in which they show that human motion is characterized by specific periodic patterns in the *similarity plot* (a 2D matrix of all pairwise image matching correlations), and describe a method for human detection by recognizing such patterns. They also use similarity plots to estimate the stride of a walking and running person, assuming a calibrated camera. Here, a person is tracked over a ground plane, and their distance traveled, D , is estimated. The number of steps N is also automatically estimated using periodic motion, which can be a non-integer number. The stride is D/N , which could be used as a biometric, though they have not conducted any study showing how useful it is as a biometric.

3 Motivation

In this section, we give the motivation for the methodology used in this paper. One method of motion-based recognition is to first explicitly extract the dynamics of points on a moving object. Consider a point $P(t) = (x(t), y(t), z(t))$ on a moving object as a function of time t (see Figure 1). The dynamics of the point can be represented by the phase plot $(P(t), dP/dt(t), \dots)$. Since we wish to recognize different types of motions (viz. gaits), it is important to know what can be determined from the projection T of $P(t)$ onto an image plane, $(u, v) = T(P)$. Under orthographic projection, and if $P(t)$ is constrained to planar motion, the object dynamics are completely preserved up to a scalar factor. That is, the phase space for the point constructed from (u, v) is identical (up to a scalar factor) to the phase space constructed from $P(t)$. However, if the motion is not constrained to a plane, then the dynamics are not preserved. Under perspective projection, the dynamics of planar and arbitrary motion are in general not preserved.

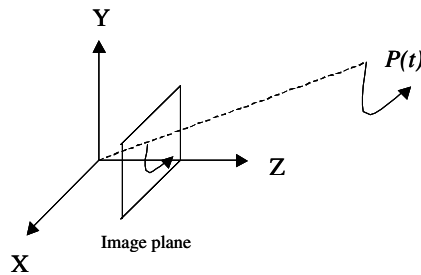


Fig. 1. Point $P(t)$ on an object moving in R^3 , imaged onto a 2D plane.

Fortunately, planar motion is an important class of motion, and includes “biological motion” [17]. In addition, if the object is sufficiently far from the camera, the camera projection becomes approximately orthographic (with scaling). In this case, and assuming we can accurately track a point $P(t)$ in the image plane, then we can completely

reconstruct the phase space of the dynamic system (up to a scalar factor). The phase space can then be used directly to classify the object motion (e.g., [7]).

In general, point correspondence is not always possible in realistic image sequences (without the use of special markers), due to occlusion boundaries, lighting changes, insufficient texture, image noise, etc. However, for classifying motions, we do not necessarily have to extract the complete dynamics of the system; qualitative measures may suffice to distinguish a class of motions from each other. In this paper, we use correspondence-free, qualitative measures for motion-based gait recognition.

4 Method

4.1 Self-Similarity Plots

Computation from an Image Sequence. Given a sequence of grey-scale images obtained from a static camera, we detect and track the moving person, extract an image template corresponding to the person's motion blob in each frame, then compute the image self-similarity plot from the obtained sequence of templates. For this, we use the method described in [10], except that we use background modeling and subtraction [12, 19] for foreground detection, since the camera is assumed to be stationary.

Moving objects are tracked in each frame based on spatial and temporal image coherence. An image template at time t , denoted by O_t , is extracted for each tracked object, consisting of the image region enclosed within the bounding box of its motion blob in the current frame. Deciding whether a moving object corresponds to a walking person is currently done based on simple shape (such as aspect ratio of the bounding box and blob size) and periodicity cues.

Once a person has been tracked for N consecutive frames, its N image templates are scaled to the same dimensions $H \times W$, as their sizes may vary due to change in camera viewpoint and segmentation errors. The image self-similarity, S , of the person is then computed as follows:

$$S(t_1, t_2) = \sum_{(x,y) \in B_{t_1}} |O_{t_1}(x, y) - O_{t_2}(x, y)|,$$

where $1 \leq t_1, t_2 \leq N$, B_{t_1} is the bounding box of the person in frame t_1 , and $O_{t_1}, O_{t_2}, \dots, O_{t_N}$ are the scaled image templates of the person. In order to account for tracking errors, we compute the minimal S' by translating over a small search radius r :

$$S'(t_1, t_2) = \min_{|dx, dy| < r} \sum_{(x,y) \in B_{t_1}} |O_{t_1}(x + dx, y + dy) - O_{t_2}(x, y)|.$$

Figure 2(b) shows a plot of S' for all combinations of t_1 and t_2 , for the two walking sequences (80 frames each) shown in Figure 2(a) (note the similarity values have been linearly scaled for visualization to the grayscale intensity range [0,255], where dark regions show more similarity).

Properties. The similarity plot, S' , of a walking person has the following properties:

1. $S'(t, t) = 0$, i.e. it has a dark main diagonal.
2. $S'(t_1, t_2) = S(t_2, t_1)$, i.e. it is symmetric along the main diagonal.
3. $S'(t_1, kp/2 + t_1) \simeq 0$, i.e. it has dark lines *parallel* to the main diagonal.
4. $S'(t_1, kp/2 - t_1) \simeq 0$, i.e. it has dark lines *perpendicular* to the main diagonal.

where $t_1, t_2 \in [1, N]$, p is the period of walking, and k is an integer. The first two properties are generally true for any similarity function (though if substantial image scaling is required, the second property may not hold). The latter two, however, are a direct consequence of the periodicity and the bilateral symmetry, respectively, of the human gait.

Figure 2(a) shows the 5 key poses over one walking cycle for two different persons; poses A and C correspond to when the person is in maximum swing (i.e. the two legs are furthest apart), and Pose B corresponds to when the two legs are together. One can easily see that the intersections of the dark lines in S' (i.e. the diagonals and cross diagonals) correspond to pose combinations AA, BB and CC, AC, and CA. Thus these intersections, which are the local minima of S' , can be used to determine the frequency and phase of walking [11].

That S' encodes the frequency and phase of gait can be explained by the fact that the similarity plot of a walking person is (approximately) a projection of the planar dynamics of the walking person when viewed sufficiently far from the camera, as previously suggested in [11]. Intuitively, this is because S' is obtained via a sequence of transformations (image projection and template matching) applied to the set of 3D points constituting the person's body. It can be shown that these transformations approximately preserve the dynamics of these points (and hence the dynamics of the gait) under certain assumptions.

4.2 Gait Classifier

As mentioned in the previous section, the similarity plot is a projection of the dynamics of the walking person that preserves the frequency and phase of the gait. The question then arises as to whether this projection preserves more detailed (higher-dimensional) aspects of gait dynamics, that capture the unique way a person walks. In other words, does a similarity plot contain sufficient information to distinguish (not necessarily uniquely) the walking gaits of different people?

To evaluate the usefulness of the self-similarity plot in characterizing and recognizing individual gaits, we propose to build a gait pattern classifier that takes an SP (self-similarity plot) as the input feature vector. For this, we take an 'eigenface' approach [35], in which we treat a similarity plot the same way that a face image is used in a face recognizer. The gist of this approach is that it extracts 'relevant information' from input feature vectors (face images or SPs) by finding the principal components of the distribution of the feature space, then applies standard pattern classification of new feature vectors in the lower-dimensional space spanned by the principal components. We use a simple non-parametric pattern classification technique for recognition. In the following, we explain the details of the proposed gait classifier.

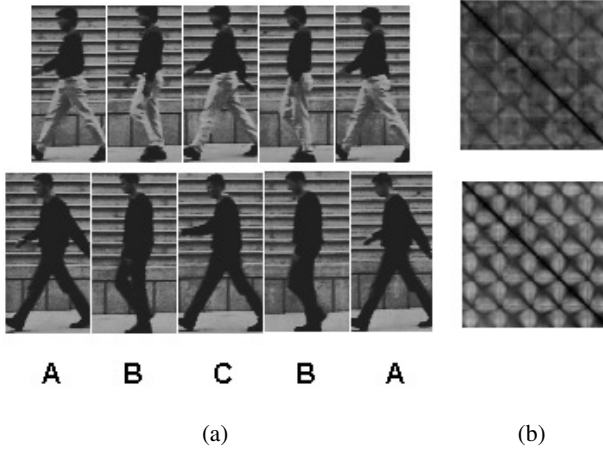


Fig. 2. (a) A few frames of the key poses in a walking person sequence, and the corresponding (b) Image self-similarity plots.

Normalizing the Input. In order to account for different walking paces and starting poses, we need to normalize the self-similarity plots so that they are phase-aligned, have the same frequency, and contain the same number of cycles. To this end, we compute the frequency and phase of each similarity plot using the method in [10]. We choose the pose corresponding to when the legs are maximally apart, i.e. poses A or C in Figure 2(a), for phase alignment.

Training the Classifier. Let S'_1, S'_1, \dots, S'_M be a given training set of M labeled (i.e. corresponding to a known person) normalized similarity plots, of size $N \times N$ each, and let s'_i be the vector of length N^2 corresponding to the i th similarity plot S'_i (obtained by concatenating all its rows). We compute the principal components [22] of the space spanned by s'_1, \dots, s'_M by computing the eigenvalue decomposition (also called Karhunen-Loeve expansion) of their covariance matrix:

$$C_s = \frac{1}{M} \sum_{i=1}^M (s'_i - \bar{s}') (s'_i - \bar{s}')^T$$

where \bar{s}' is the simple mean of all training vectors s'_1, \dots, s'_M . This can be efficiently computed in $O(M)$ time (instead of the brute force $O(N^2)$) [65].

We then consider the space spanned by the n most significant eigenvectors, u_1, \dots, u_n , that account for 90% of the variation in the training SPs [1]. We denote this space the

¹ According to the theory of PCA, if $\lambda_1, \dots, \lambda_n$ are the n largest eigenvalues, then the space spanned by their corresponding eigenvectors account for $\sum_{i=1}^n \lambda_i / \text{trace}(C_s)$ of the total variation in the original feature vectors.

Eigengait. Hence each training vector s'_i can be sufficiently approximated by a n -dimensional vector w_i obtained by projecting it onto the Eigengait, i.e. $w_i \equiv \sum_{j=1}^n u_j^T \cdot s'_i$. Furthermore, assuming that the training vectors are representative of the variation in the entire feature space, then any new feature vector can be similarly approximated by a point in Eigengait space.

Classification. Gait recognition now reduces to a standard pattern classification in a n -dimensional Eigengait space. The advantage of doing pattern classification in this space is not only that n is typically much smaller than N^2 and M , but also that it contains less unwanted variation (i.e. random noise)¹ and hence provides better separability of the feature vectors, or SPs.

Given a new SP (corresponding to an unknown person), the procedure for recognizing it is to first convert it to a N^2 -vector, map it to a point in Eigengait, find the k closest training points to it, then decide its class (or label) via the *k-nearest neighbor rule* [5, 51].

5 Experiments

To evaluate our method, we build the gait classifier described above using k-nearest neighbor classification and the gait data set of Little and Boyd [23]. We use the leave-one-out cross-validation to obtain a statistically accurate estimate of the recognition rate [36, 31].

The data set consists of 40 image sequences and six different subjects (7 sequences per person except for the 5th person). Figure 3 shows all six subjects overlaid at once on the background image.

Since the camera is static we used median filtering to recover the background image. Templates of the moving person were extracted from each image by computing the difference of the image and the background and subsequently applying a threshold as well as morphological operations to clean up noise. This simple method for person detection and tracking was sufficient because the background is static and each sequence only contains one moving person. Absolute correlation was used to compute the self-similarity plot for each of the 40 template sequences. For temporal normalization (since the image sequences were of varying lengths and the persons had different gait cycle lengths), the similarity plots were cropped and scaled so that they all contained 4 gait cycles starting on the same phase, and are of size 64x64. Figure 4 shows examples of these normalized similarity plots, where each column of three plots corresponds to one person.

Since our data set is relatively small, we use the leave-one-out cross-validation method. The leave-one-out error rate estimator is known to be an (almost) unbiased estimator of the true error rate of the classifier. Hence, out of the 40 similarity plots, we build (or train) our classifier on all but one of the samples, test the classifier on the sample missed (or left out), and record the classification result. This is repeated 40 times, leaving out each of the 40 samples in turn. The recognition rate is then obtained as the ratio of the number of correctly classified test samples out of the total 40.

¹ Assuming data variation is much larger than noise variation.

The classifier is built simply by storing the training vectors as points in Eigengait space, and the test sample is classified by determining its k -nearest neighbor with $k = 5$, using the Euclidian distance as a distance metric and simple majority as a decision rule. The recognition rate thus obtained is 0.93 (37 out of 40).

To visualize how well separated the gaits of the 6 people are, we applied PCA to all 40 SPs. Figure 5 shows all 40 SPs projected onto the 3 most significant eigenvectors thus obtained. Each closed contour encloses the samples (points) corresponding to one person.



Fig. 3. The six people contained in the test sequences, overlaid on the background image.

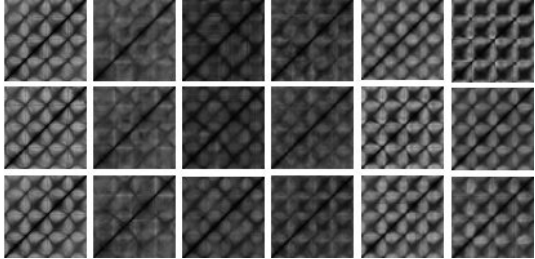


Fig. 4. Normalized self-similarity plots (columns correspond to a single person).

6 Conclusion and Future Work

In this paper, we have used a correspondence-free motion-based method to recognize the gaits of a small population (6) of people. While the results are promising, more evaluation of the method needs to be done. Future studies include larger test populations (20-100 people) and images taken from multiple view points (not just parallel to the image plane 1). In addition, image sequences of the same individual need to be acquired in different lighting conditions, and with various types of clothing.

³ The Keck Laboratory 6 will be utilized to acquire multiview (up to 64) images sequences of a person walking.

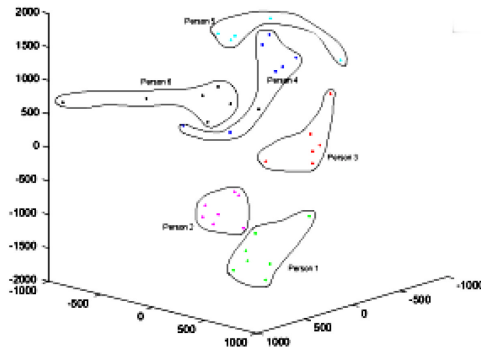


Fig. 5. Similarity plots projected onto the space spanned by the three most dominant eigenvectors.

Finally, we are working to combine the results of this correspondence-free gait recognition method with more feature-oriented methods, such as stride and height estimation.

Acknowledgment

We would like to thank Dr. Jeffrey E. Boyd of the Department of Computer Science at the University of Calgary, Canada, for providing the gait data we used in testing. The support of DARPA (Human ID project, grant No. 5-28944) is also gratefully acknowledged.

References

- [1] J. K. Aggarwal and Q. Cai, "Human motion analysis: a review," in *Proc. of IEEE Computer Society Workshop on Motion of Non-Rigid and Articulated Objects*, 1997.
- [2] K. Akita, "Image Sequence Analysis of Real World Human Motion," Vol. 17, No. 1, pp. 73–8, 1984.
- [3] C. Barclay, J. Cutting, and L. Kozlowski, "Temporal and Spatial Factors in Gait Perception that Influence Gender Recognition," *Perception and Psychophysics*, Vol. 23, No. 2, pp. 145–152, 1978.
- [4] J. Bigun, G. Chollet, and G. Borgefors, *Audio- and Video-based Biometric Person Authentication*, Springer, 1997.
- [5] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford: Clarendon Press, 1995.
- [6] E. Borovikov, R. Cutler, T. Horprasert, and L. Davis, "Multi-perspective Analysis of Human Actions," 1999.
- [7] L. W. Campbell and A. Bobick, "Recognition of Human Body Motion Using Phase Space Constraints," 1995.
- [8] C. Cedras and M. Shah, "A survey of motion analysis from moving light displays," pp. 214–221, 1994.

- [9] D. Cunado, M. Nixon, and J. Carter, "Using Gait as a Biometric, via Phase-Weighted Magnitude Spectra," in *Proceedings of 1st Int. Conf. on Audio- and Video-Based Biometric Person Authentication*, pp. 95–102, 1997.
- [10] R. Cutler and L. Davis, "Robust Real-time Periodic Motion Detection, Analysis and Applications," Vol. 13, No. 2, pp. 129–155, 2000.
- [11] J. Cutting and L. Kozlowski, "Recognizing Friends by Their Walk: Gait Perception Without Familiarity Cues," *Bulletin Psychonomic Soc.*, Vol. 9, No. 5, pp. 353–356, 1977.
- [12] J. W. Davis and A. F. Bobick, "The representation and recognition of action using temporal templates," pp. 928–934, 1997.
- [13] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *Proceedings of International Conference on Computer Vision*, 2000.
- [14] D. Gavrilu, "The visual analysis of human movement: a survey," Vol. 73, pp. 82–98, January 1999.
- [15] D. M. Gavrilu and L. Davis, "Towards 3-D Model-based Tracking and Recognition of Human Movement: a Multi-View Approach," (Zurich, Switzerland), 1995.
- [16] Q. He and C. Debrunner, "Individual Recognition from Periodic Activity Using Hidden Markov Models," in *IEEE Workshop on Human Motion*, 2000.
- [17] D. Hoffman and B. Flinchbaugh, "The interpretation of biological motion," *Biological Cybernetics*, 1982.
- [18] D. Hogg, "Model-based vision: a program to see a walking person," *Image and Vision Computing*, Vol. 1, No. 1, 1983.
- [19] T. Horprasert, D. Harwood, and L. Davis, "A Robust Background Subtraction and Shadow Detection," 2000.
- [20] P. S. Huang, C. J. Harris, and M. S. Nixon, "Comparing Different Template Features for Recognizing People by their Gait," in *BMVC*, 1998.
- [21] G. Johansson, "Visual Motion Perception," *Scientific American*, pp. 75–88, June 1975.
- [22] I. T. Joliffe, *Principal Component Analysis*, Springer-Verlag, 1986.
- [23] J. Little and J. Boyd, "Recognizing people by their gait: the shape of motion," *Videre*, Vol. 1, No. 2, 1998.
- [24] F. Liu and R. Picard, "Finding periodicity in space and time," pp. 376–383, January 1998.
- [25] K. Luttgens and K. Wells, *Kinesiology: Scientific Basis of Human Motion*, Saunders College Publishing, 7th ed., 1982.
- [26] D. Meyer, J. Pösl, and H. Niemann, "Gait Classification with HMMs for Trajectories of Body Parts Extracted by Mixture Densities," in *BMVC*, pp. 459–468, 1998.
- [27] H. Murase and R. Sakai, "Moving object recognition in eigenspace representation: gait analysis and lip reading," Vol. 17, pp. 155–162, 1996.
- [28] M. Murray, "Gait as a total pattern of movement," *American Journal of Physical Medicine*, Vol. 46, No. 1, pp. 290–332, 1967.
- [29] S. Niyogi and E. Adelson, "Analyzing and recognizing walking figures in XYT," pp. 469–474, 1994.
- [30] R. Polana and R. Nelson, "Detection and Recognition of Periodic, Non-rigid Motion," Vol. 23, pp. 261–282, June/July 1997.
- [31] B. Ripley, *Pattern Recognition and Neural Networks*, Cambridge: Cambridge University Press, 1996.
- [32] K. Rohr, "Towards model-based recognition of human movements in image sequences," in *CVGIP*, vol. 59, 1994.
- [33] Y. Song, X. Feng, and P. Perona, "Towards Detection of Human Motion," 2000.
- [34] P. Tsai, M. Shah, K. Keiter, and T. Kasparis, "Cyclic Motion Detection for Motion-based Recognition," Vol. 27, No. 12, pp. 1591–1603, 1994.
- [35] M. Turk and A. Pentland, "Face Recognition using Eigenfaces," 1991.
- [36] S. Weiss and C. Kulikowski, *Computer Systems that Learn*, Morgan Kaufman, 1991.

Visual Categorization of Children and Adult Walking Styles

James W. Davis

Motion Recognition Laboratory
Dept. of Computer and Information Science
Ohio State University, Columbus, OH 43210 USA
`jwdavis@cis.ohio-state.edu`

Abstract. We present an approach for visual discrimination of children from adults in video using characteristic regularities present in their locomotion patterns. The framework employs computer vision to analyze correlated, scale invariant locomotion properties for classifying different styles of walking. Male and female subjects for the experiments include six children (3–5 yrs) and nine adults (30–52 yrs). For the analysis, we coordinate a minimalist point-representation of the human body with a space-time analysis of head and ankle trajectories to characterize the modality. Together the properties of relative stride length and stride frequency are shown to clearly differentiate children from adult walkers. The highly correlated log-linear relationships for the stride properties are exploited to reduce the categorization problem to a linear discrimination task. Using a trained two-class linear perceptron, we were able to achieve a correct classification rate of 93–95% on our dataset. Our approach emphasizing the natural modal behavior in human motion offers a useful and general methodology as the basis for designing efficient motion recognition systems using limited visual features.

1 Introduction

We can easily perceive a person walking in the distance simply from viewing the characteristic pattern of human motion. We can even identify a close friend from the way he or she walks. Remote-sensing computer monitoring and security systems designed to visually interpret our moving world will also require similar abilities to recognize human movements. In this paper we describe a categorical recognition system that addresses one of the most fundamental classes associated with human movement — locomotion. Human locomotion is subject to a variety of physical and dynamical constraints, which together create a tight region – or “mode” – in some multi-dimensional feature space. Our belief is that such correlated *visual* properties in human locomotion can be used within computer vision systems for the reliable classification of walking people. In support of this claim, we describe a categorical vision approach for distinguishing children from adults using the inherent modal nature associated with the properties of relative stride length and stride frequency during locomotion.

Much research has appeared in the computer vision literature pertaining to people walking, including several model-based tracking approaches [3, 11, 12, 4],

static-based pedestrian detection methods [12, 5], and trajectory-based recognition systems [10, 11, 4]. Our approach differs from the aforementioned recognition methods in that we seek dynamical correlations between movement features to infer *categories* of people (e.g. child, adult) from their walking motions. The applied significance of the child-adult recognition paradigm impacts those automated visual surveillance and monitoring systems interested in identifying child and adult behaviors. Monitoring systems in locations such as shopping malls and airports place special importance on the detection of a lone or wandering child. Also, a smart car able to detect crossing pedestrians [3] would have a further safety advantage to avoid accidents if the car were able to have a specific awareness of children walking nearby.

2 Human Locomotion Features

Measurements of time and distance for each walking cycle represent the most basic descriptions that determine a particular gait [9]. Such parameters include stride length (distance between footfalls of the same foot), stature (body height), and cycle time (time of one complete cycle of one leg).

A commonly used variable in describing locomotion is *relative stride* L' , calculated by normalizing the person's stride length by stature. The result is a dimensionless number with no issue of inches or pixels, and thus it can be used to compare the relative spatial configuration of children and adults. Another descriptive temporal feature is *stride frequency* f (strides/min), computed from the inverse of the cycle time. We list the locomotion features in Table 1 for convenience. These stride properties are visual features and can therefore be used, in part, by computer vision systems for the analysis of locomotion. As we will show, two distinctive modal relationships for the conjunction of the stride-based properties can be used to classify child and adult locomotion.

Table 1. Fundamental locomotion features.

Description	Symbol	Units	Obtained
Stride length	L	pixels or inches	measured
Stature	S	pixels or inches	measured
Leg cycle time	T_c	sec.	measured
Relative stride	L'	—	L/S
Stride frequency	f	min.^{-1}	$60/T_c$

3 Methods

Male and female subjects having normal gaits used for the experiments included nine adults 30–52 years old and six children 3–5 years old. This particular age

range for children was motivated by the reported biomechanical difference in the walking style of children 3–5 years old as compared with adolescents and adults¹.

To calculate the proposed stride-based features (L^0 , f) for the walkers, only the locations of the head and feet in each video frame are required. These extremity points are much more attainable than either joint angles or limb lengths/poses from images. We opted to initially record subjects walking while adorned with reflective markers on the head and ankles for the experiments. We are currently examining automatic methods (similar to [17, 12, 9]) for identifying these body locations from multiple viewpoints in un-marked video.

The imaging configuration positioned an infrared video camcorder (Sony CCD-TRV87) at a distance of 12.5 feet fronto-parallel to the walker (at 3-foot elevation for adults and 2-foot elevation for children). The adult subjects were recorded walking on a motorized treadmill (See Fig. 1.a) at speeds ranging from 1.5–4.5 MPH, increasing in 0.2 MPH increments (individuals had different maximum speeds). Treadmill and overground walking strides are not significantly different for the major range of the walking speeds [2]. The six child subjects were recorded walking at different speeds across a room in front of the camera (See Fig. 1.b). Due to the nature of using young children as experimental subjects, only those natural-looking walking sequences were retained. The trials were recorded onto a VCR, and later digitized and de-interlaced to achieve a faster frame-rate of 60 Hz at 320 × 240 resolution.

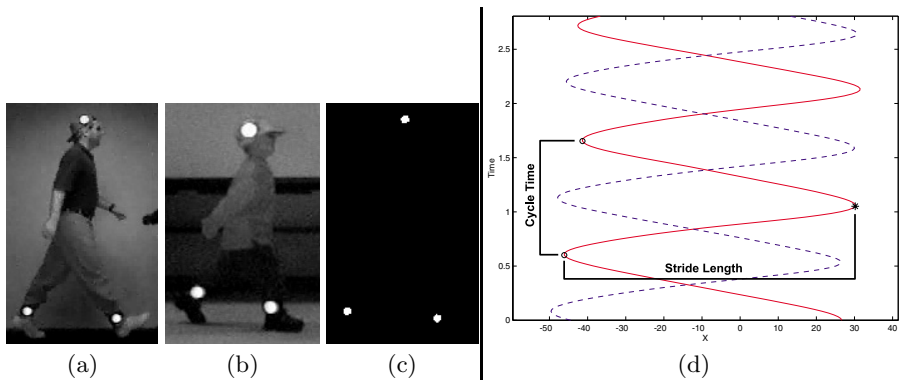


Fig. 1. Walkers and point-light motions. (a) Adult walking on a treadmill at 4.5 MPH. (b) Child walking across a room. (c) Thresholding image in (a) produces point-lights to be tracked. (d) Ankle x-trajectories with stride length and cycle time noted.

¹ For children 3–5 years old, their leg swing time remains relatively constant over various walking speeds, while for adults the swing time is negatively correlated with increasing walking speed [1].

3.1 Automatic Feature Extraction

We first threshold the images in each walking sequence to highlight the reflective markers on the head and ankles (See Fig. 1c). After using a region-growing algorithm to identify the three centroids, trajectories were automatically created using the point tracking and correspondence approach of [13] which minimizes a *proximal uniformity function* δ for points q, r using three frames X^{t-1}, X^t, X^{t+1} with

$$\delta_{qr}^{t+1}(X) = \frac{\| \overline{X_q^{t-1} X_q^t} - \overline{X_q^t X_r^{t+1}} \|}{\sum_i \sum_j \| \overline{X_i^{t-1} X_i^t} - \overline{X_i^t X_j^{t+1}} \|} + \frac{\| \overline{X_q^t X_r^{t+1}} \|}{\sum_i \sum_j \| \overline{X_i^t X_j^{t+1}} \|} \quad (1)$$

where the first and second terms represent smoothness and proximity constraints, respectively. We then removed the translation component in the resulting trajectories using the horizontal location of the head point as a reference in each frame, followed by lowpass filtering.

To calculate the relative stride ($L' = L/S$) for rightward walking (the method is similar for leftward walking), the stride length L is computed as the pixel distance between consecutive minima and maxima locations in an ankle x-trajectory (See Fig. 1d); the image stature S is calculated as the pixel distance from the head point to the ankle point of the support leg at criss-crossing locations. The cycle time for the stride frequency ($f = 60/T_c$) can be determined by measuring the time between two consecutive minima in an ankle x-trajectory (See Fig. 1d). These features were automatically extracted for each step cycle for each ankle of the walkers in our dataset. With these features, we can now compare the walking styles of the children and adults.

4 Results

The computed ranges of stride frequencies for the children and adults were 55.3–89.8 strides/min and 36.7–73.3 strides/min, respectively. The relative strides for the children and adults were in the range 0.27–0.55, hence they share the same relative extent of spatial stride configurations. In Fig. 2a we present a log-linear plot of relative stride vs. stride frequency as calculated from the children and adults walking at different speeds. Sub-spaces (modes) of child and adult locomotion are quite apparent in the data. A general interpretation of this plot is that whenever a child has the same (or larger) relative stride configuration as an adult, the child has a larger stride frequency.

With such strong and definitive modal sub-spaces, a simple linear decision boundary is all that is required to separate the two classes to a high degree. The data appears non-Gaussian, hence for classifying categories c1 and c2 we used a two-class linear perceptron discriminator having the general form

$$d(\mathbf{x}) = \sum_1^n w_i x_i + w_{n+1} = 0, \quad \text{with} \quad \sum_1^n w_i x_i \begin{matrix} \text{c1} \\ > \\ \text{c2} \end{matrix} - w_{n+1}. \quad (2)$$

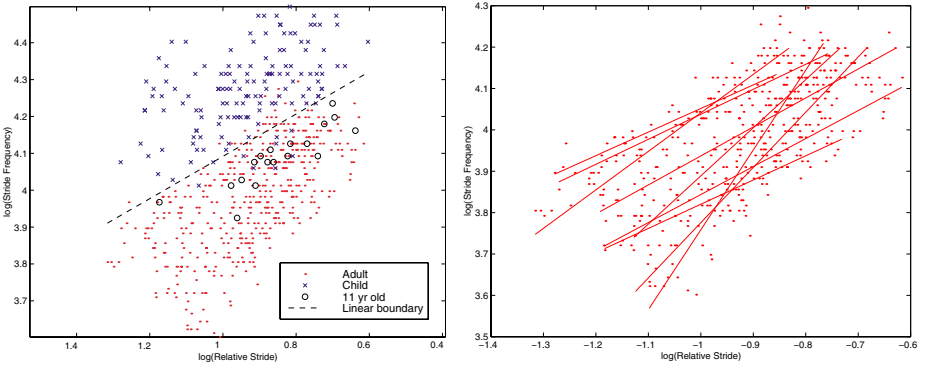


Fig. 2. (a) Walking modes for relative stride vs. stride frequency in children and adults (natural log values). A linear classification boundary is shown separating the two categories. The walking motions of an 11 year old reside entirely in the adult category. (b) Linear correlations for adult data points in (a), where each line represents an individual adult ($\mu = 0.0232 \pm 0.0287$ SD).

A perceptron neural network was trained with all the walking examples resulting in a network having output O according to

$$O = \begin{cases} Adult & \text{if } 4376 \leq \ln(L) \leq 7624 \text{ and } \ln(f) > 0.355718 \\ Child & \text{if } 4376 \leq \ln(L) \leq 7624 \text{ and } \ln(f) < 0.355718 \end{cases} \quad (3)$$

determined after 30K epochs using a Matlab Neural Network Toolbox implementation (See boundary in Fig. 2a). When the dataset is classified using this discriminator, we receive 95% correct classification for the adults and 93% correct classification for the children. Three older children 5–6 years old were also tested and shown to have more motions associated with the adult category ($\approx 30\%$), suggesting the beginnings of a change in walking style. When an 11 year old was examined, his motions existed entirely within the adult category of locomotion (See Fig. 2a).

The varying slope and positioning of the linear correlations for each adult over various speeds, as shown in Fig. 2b, clearly show that there is no simple relationship of relative stride and stride frequency to stature. As individual mode lines are more unique than single points (many lines intersect), we could however use the individual modes to assess whether an observed motion is *consistent* with a hypothesized identity. To help confirm identity, one could calculate the stride properties for the person and compute the distance of that point to the mode of the proposed individual i using $D_i = \frac{1}{2} \ln\left(\frac{L}{L_i}\right) + \frac{1}{2} \ln\left(\frac{f}{f_i}\right) + \frac{1}{2} \sqrt{\frac{L_i^2}{L^2} + \frac{f_i^2}{f^2}}$ and verify that the motion is within tolerance. Though it is unlikely that stride-based features alone can be used for person identification, they may however be applicable to person authentication.

5 Summary

We presented an approach for the visual discrimination of children (3–5 years old) from adults using stride-based properties of their walking style. Trajectories of marked head and ankle positions for six children and nine adults were used to compute the relative stride and stride frequency for each walker at different speeds. The distinction between child and adult for these features is quite strong and reduces the task of categorization to a linear discrimination test. Using a trained two-class linear perceptron, we were able to achieve a correct classification rate of 93–95% for our dataset. Given that only two motion features were used to characterize and differentiate children from adults, the result is quite encouraging. The use of natural modes as a means of visual categorization provides a useful bottom-up framework for the classification and recognition of humans in motion.

References

1. A. Baumberg and D. Hogg. Learning flexible models from image sequences. In *Proc. European Conf. Comp. Vis.*, pages 299–308, 1994.
2. W. Boda, W. Tapp, and T. Findley. Biomechanical comparison of treadmill and overground walking. In *Proc. Can. Soc. for Biomech.*, pages 88–89, 1994.
3. C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Proc. Comp. Vis. and Pattern Rec.*, pages 8–15, 1998.
4. I. Chang and C. Huang. The model-based human body motion analysis system. *Image and Vision Comp.*, 18(14):1067–1083, 2000.
5. D. Gavrilu. Pedestrian detection from a moving vehicle. In *Proc. European Conf. Comp. Vis.*, pages 37–49, 2000.
6. D. Grieve and R. Gear. The relationship between length of stride, step frequency, time of swing and speed of walking for children and adults. *Ergonomics*, 5(9):379–399, 1966.
7. I. Haritaoglu, D. Harwood, and L. Davis. W4: Who? When? Where? What? A real time system for detecting and tracking people. In *Proc. Int. Conf. Auto. Face and Gesture Recog.*, pages 222–227, 1998.
8. D. Hogg. Model-based vision: a program to see a walking person. *Image and Vision Comp.*, 1(1):5–20, 1983.
9. V. Inman, H. Ralston, and F. Todd. *Human Walking*. Williams & Wilkins, Baltimore, 1981.
10. J. Little and J. Boyd. Describing motion for recognition. In *Proc. Symp. Comp. Vis.*, pages 235–240. IEEE, 1995.
11. S. Niyogi and E. Adelson. Analyzing and recognizing walking figures in XYT. In *Proc. Comp. Vis. and Pattern Rec.*, pages 469–474. IEEE, 1994.
12. M. Oren, C. Papageorgiour, P. Sinha, E. Osuma, and T. Poggio. Pedestrian detection using wavelet templates. In *Proc. Comp. Vis. and Pattern Rec.*, pages 193–99. IEEE, 1997.
13. K. Rangarajan and M. Shah. Establishing motion correspondence. *Comp. Vis., Graph., and Img. Proc.*, 54(1):56–73, 1991.
14. K. Rohr. Towards model-based recognition of human movements in image sequences. *Comp. Vis., Graph., and Img. Proc.*, 59(1):94–115, 1994.
15. C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfunder: real-time tracking of the human body. *IEEE Trans. Patt. Analy. and Mach. Intell.*, 19(7):780–785, 1997.

A Multi-view Method for Gait Recognition Using Static Body Parameters

Amos Y. Johnson¹ and Aaron F. Bobick²

¹ Electrical and Computer Engineering
Georgia Tech, Atlanta, GA 30332
amos@cc.gatech.edu

² GVU Center/College of Computing
Georgia Tech, Atlanta, GA 30332
afb@cc.gatech.edu

Abstract. A multi-view gait recognition method using recovered static body parameters of subjects is presented; we refer to these parameters as *activity-specific biometrics*. Our data consists of 18 subjects walking at both an angled and frontal-parallel view with respect to the camera. When only considering data from a single view, subjects are easily discriminated; however, discrimination decreases when data across views are considered. To compare between views, we use ground truth motion-capture data of a reference subject to find scale factors that can transform data from different views into a common frame (“walking-space”). Instead of reporting percent correct from a limited database, we report our results using an expected confusion metric that allows us to predict how our static body parameters filter identity in a large population: lower confusion yields higher expected discrimination power. We show that using motion-capture data to adjust vision data of different views to a common reference frame, we can get achieve expected confusions rates on the order of 6%.

1 Introduction

Automatic gait recognition is new emerging research field with only a few researched techniques. It has the advantage of being unobtrusive because body-invading equipment is not needed to capture gait information. From a surveillance perspective, gait recognition is an attractive modality because it may be performed at a distance, surreptitiously.

In this paper we present a gait recognition technique that identifies people based on static body parameters recovered during the walking action across multiple views. The hope is that because these parameters are directly related to the three-dimensional structure of the person they will be less sensitive to error introduced by variation in view angle. Also, instead of reporting percent correct (or recognition rates) in a limited database of subjects, we derive an expected confusion metric that allows us to predict how well a given feature vector will filter identity over a large population.

1.1 Previous Work

Perhaps the first papers in the area of gait recognition comes from the Psychology field. Kozlowski and Cutting [34] determined that people could identify other people base solely on gait information. Stevenage, Nixon, and Vince [42] extended the works by exploring the limits of human ability to identify other humans by gait under various viewing conditions.

Automatic gait-recognition techniques can be roughly divided into model-free and model-based approaches. Model-free approaches [7,9,10] only analyze the shape or motion a subject makes as they walk, and the features recovered from the shape and motion are used for recognition. Model-based techniques either model the person [11] or model the walk of the person [3]. In person models, a body model is fit to the person in every frame of the walking sequence, and parameters (i.e. angular velocity, trajectory) are measured on the body model as the model deforms over the walking sequence. In walking models, a model of how the person moves is created, and the parameters of the model are learned for every person.

Because of the recency of the field, most gait recognition approaches only analyze gait from the side view without exploring the variation in gait measurements caused by differing view angles. Also, subject databases used for testing are typically small (often less than ten people); however, even though subject databases are small, results are reported as percent correct. That is, on how many trials could the system correctly recognize the individual by choosing its best match. Such a result gives little insight as to how the technique might scale when the database contains hundreds or thousands or more people.

1.2 Our Approach

Our approach to the study of gait recognition attempts to overcome these deficiencies by taking three fundamentally different steps than previous researchers.

First, we develop a gait-recognition method that recovers static body and stride parameters of subjects as they walk. Our technique does not directly analyze the dynamic gait patterns, but uses the action of walking to extract relative body parameters. This method is an example of what we call *activity-specific biometrics*. That is, we develop a method of extracting some identifying properties of an individual or of an individual's behavior that is only applicable when a person is performing that specific action. Gait is a excellent example of this approach because not only do people walk much of the time making the data accessible, but also many techniques for activity recognition are able to detect when someone is walking. Examples include the motion-history method of Bobick and Davis [3] and even the walker identification method of Nyogi and Adelson [14].

Second, we develop a walking-space adjustment method that allows for the identification of a subject walking at different view angles to the viewing plane of a camera. Our static body parameters are related to the three-dimensional structure of the body so they are less sensitive to variation in view angle. However, because of projection into an image, static body parameters recovered from different views need to be transformed to a common frame.

Finally, as opposed to reporting percent correct, we will establish the uncertainty reduction that occurs when a measurement is taken. For a given measured property, we establish the spread of the density of the overall population. To do so requires only enough subjects such that our estimate of the population density approaches some stable value. It is with respect to that density that we determine the expected variation in the measurement when applied to a given individual.

The remainder of this paper is as follows: we describe the expected confusion metric we used to evaluate our technique, present the gait-recognition method, and describe how to convert the different view-angle spaces to a common walking-space. Last, we will assess the performance of our technique using the expected confusion metric.

2 Expected Confusion

As mentioned our goal is not to report a percent correct of identification. To do so requires us to have an extensive database of thousands of individuals being observed under a variety of conditions. Rather, our goal is to characterize a particular measurement as to how much it reduces the uncertainty of identity after the measurement is taken.

Many approaches are possible. Each entails first estimating the probability density of a given property vector \mathbf{x} for an entire population $P_p(\mathbf{x})$. Next we must estimate the uncertainty of that property for a given individual once the measurement is known $P_I(\mathbf{x}|\eta = \mathbf{x}_0)$ (interpreted as what is the probability density of the true value of the property \mathbf{x} after the measurement η is taken). Finally, we need to express the average reduction in uncertainty or the remaining confusion that results after having taken the measurement.

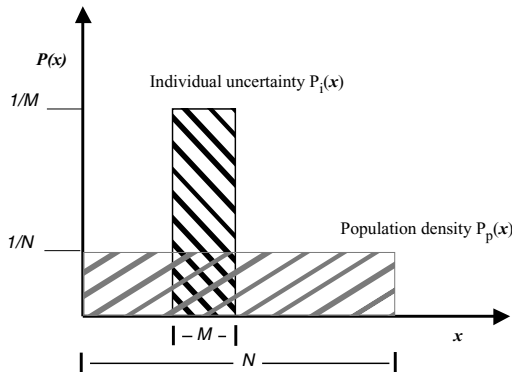


Fig. 1. Uniform probability illustration of how the density of the overall population compares to the the individual uncertainty after the measurement is taken. In this case the remaining confusion — the percentage of the population that could have given rise to the measurement — is M/N .

Information theory argues for a mutual information [2] measure:

$$I(\mathbf{X}; \mathbf{Y}) = H(\mathbf{X}) - H(\mathbf{X}|\mathbf{Y}). \quad (1)$$

where $H(\mathbf{X})$ is the entropy of a random variable \mathbf{X} defined by

$$H(\mathbf{X}) = - \int_x p(x) \ln p(x),$$

and $H(\mathbf{X}|\mathbf{Y})$ is the conditional entropy of a random variable \mathbf{X} given another random variable \mathbf{Y} defined by:

$$H(\mathbf{X}|\mathbf{Y}) = - \int_{x,y} p(x,y) \ln p(x|y).$$

For our case the random variable \mathbf{X} is the underlying property (of identity) of an individual before a measurement is taken and is represented by the population density of the particular metric used for identification. The random variable \mathbf{Y} is an actual measurement retrieved from an individual and is represented by a distribution of the individual variation of an identity measurement. Given these definitions, the uncertainty of the property (of identity) of the individual given a specific measurement, $H(\mathbf{X}|\mathbf{Y})$, is just the uncertainty of the measurement, $H(\mathbf{Y})$. Therefore the mutual information reduces to:

$$I(\mathbf{X}; \mathbf{Y}) \equiv H(\mathbf{X}) - H(\mathbf{Y}). \quad (2)$$

Since the goal of gait recognition is filtering human identity this derivation of mutual information is representative of filtering identity. However, we believe that a better assessment (and comparable to mutual information) of a metric's ability to filter identity is the expected value of the percentage of the population eliminated after the measurement is taken. This is illustrated in Figure 1. Using a uniform density for illustration we let the density of the feature in the population P_p be $1/N$ in the interval $[0, N]$. The individual density P_i is much narrower, being uniform in $[x_0 - M/2, x_0 + M/2]$. The confusion that remains is the area of the density P_p that lies under P_i . In this case, that confusion ratio is M/N .

An analogous measure can be derived for the Gaussian case under the assumption that the population density σ_p is much greater than the individual variation σ_i . In that case the expected confusion is simply the ratio σ_i/σ_p , the ratio of standard deviation of the uncertainty after measurement to that before the measurement is taken. Note that if the negative natural logarithm of this is taken we get:

$$- \ln\left(\frac{\sigma_i}{\sigma_p}\right) = \ln \sigma_p - \ln \sigma_i, \quad (3)$$

we arrive at an expression that is the mutual information (of two 1D Gaussian distributions) from Equation 2. For the multidimensional Gaussian case, the result is

$$\text{Expected Confusion} = \frac{|\Sigma_i|^{1/2}}{|\Sigma_p|^{1/2}}. \quad (4)$$

This quantity is the ratio of the individual variation volume over the population volume. These are volumes of equal probability hyper-ellipsoids as defined by the Gaussian densities. See [11] for complete proof.

3 Gait Recognition Method

Using a single camera with the viewing plane perpendicular to the ground plane, 18 subjects walked in an open indoor-space at two view angles: a 45° path (angle-view) toward the camera, and a frontal-parallel path (side-view) in relation to the viewing plane of the camera. The side-view data was captured at two different depths, 3.9 meters and 8.3 meters from camera. These three viewing conditions are used to evaluate our multi-view technique.

In the following subsections we explain our body part labeling technique and our depth compensation method. The body part labeling technique is used to arrive at the static body parameters of a subject. Depth compensation is used to compensate for depth changes of the subject as the walk. Lastly, before stating the results of the experiments, we present the static body parameters used and how we adjust

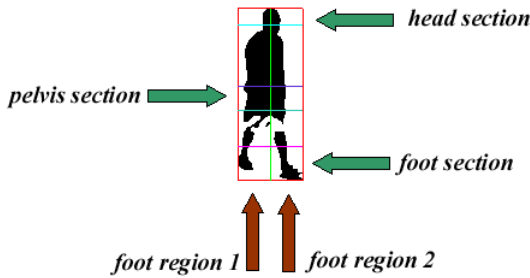


Fig. 2. Automatic segmenting of the body silhouette into regions.

3.1 Body Part Labeling

Body parts are labeled by analyzing the binary silhouette of the subject in each video frame. Silhouettes are created by background subtraction using a static background frame. A series of morphological operations are applied to the resulting images to reduce noise. Once a silhouette is generated, a bounding box is placed around the silhouette and divided into three sections – head section, pelvis section, and foot section (see Figure 2) – of predefined sizes similar to the body part labeling method in [6]. The head is found by finding the centroid of the pixels located in the head section. The pelvis is contained in pelvis section, and is the centroid of this section. The foot section houses the lower legs and feet, and is further sub-divided down the center into foot region 1 and foot region 2. Within foot region 1 and foot region 2, the distance (L2 norm) between each pixel and the previously discovered head location is calculated. The pixel location with the highest distance in each region is labeled foot 1 and foot 2. The labels do not distinguish between left and right foot because it is not necessary in

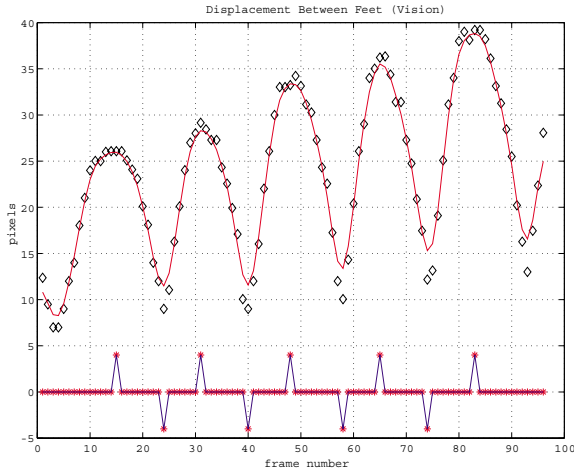


Fig. 3. The distance between the two feet as measured in pixels. The value increases as the subject approaches the camera. The curve is an average that underestimates the value of the peak but well localizes them. The lower trace indicates the maximal and minimal separations.

our technique. This method of body part labeling allows for imperfections in the silhouette due to noisy background subtraction by using local body part searches and placing soft constraints on body part locations.

3.2 Depth Compensation

The static body parameters used for identification will be a set of distances between the body parts locations, and the distances will be measured in pixels; however, a conversion factor from pixels to centimeters is needed for the possible depth locations of the subjects in the video footage. We have created a depth compensation method to handle this situation by having a subject of known height walk at an angle towards the camera. At the points of minimal separation of the subject’s feet (see Figure 3), the system measures the height (this is taken to be the height of the bounding box around the subject) of the subject in pixels at that location on the ground plane. The minimal point represents the time instances where the subject is at his or her maximal height during the walking action. A conversion factor from pixels to centimeters at each known location on the ground (taken to be the lower y -value of the bounding box) is calculated by:

$$\text{Conversion Factor} = \frac{\text{known height (centimeters)}}{\text{measured height (pixels)}}. \quad (5)$$

To extrapolate the conversion factors for the other unknown locations on the ground plane a hyperbola is fit to the known conversion factors. Assuming a world coordinate system located at the camera focal point and an image plane perpendicular to ground plane, using perspective projection we derive a

conversion factor hyperbola,

$$\text{Conversion Factor}(y_b) = \frac{A}{B - y_b}, \quad (6)$$

where A is the vertical distance between the ground and focal point times the focal length, B is the optical center (y component) of the image plus a residual (if the image plane is not exactly perpendicular to the ground), and y_b is the current y -location of the subject's feet. We implicitly estimate the parameters A and B by fitting the conversion factor hyperbola (Equation 6) to the known locations of the subject and the required conversion factors needed to covert the measured height in pixels to its known height in centimeters (see Figure 4).

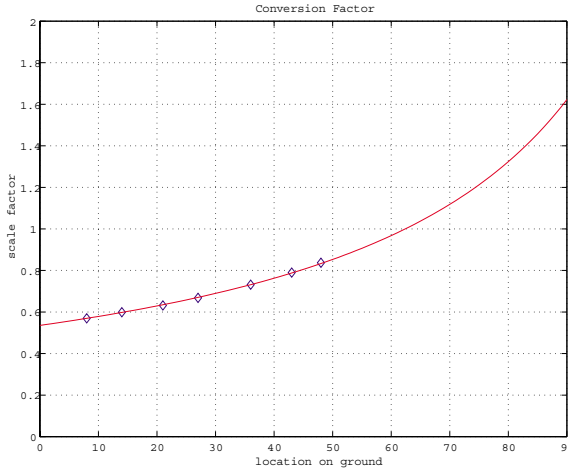


Fig. 4. Hyperbola fit to the data relating the lower y position of the bounding box to the required conversion factor. The data points are generated by observing a subject of known height walking in the space.

3.3 Static Body Parameters

After body labeling and depth compensation, a 4D-walk vector (which are the static body parameters) is computed as (see Figure 5):

- d_1 : The height of the bounding box around the silhouette.
- d_2 : The distance (L2 norm) between the head and pelvis locations.
- d_3 : The maximum value of the distance between the pelvis and left foot location, and the distance between the pelvis and right foot location.
- d_4 : The distance between the left and right foot.

These distances are concatenated to form a 4D-walk vector $\mathbf{w} = \langle d_1, d_2, d_3, d_4 \rangle$, and they are only measured when the subjects' feet are maximally spread during the walking action. As subjects walk they have multiple maximally spread

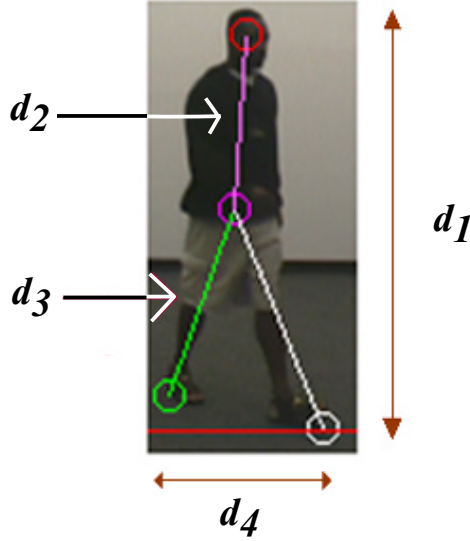


Fig. 5. The 4 static body parameters: $\mathbf{w} = \langle d_1, d_2, d_3, d_4 \rangle$.

points (see Figure 3), and the mean value of \mathbf{w} at these points is found to generate one walk vector per walking sequence. Measurements are taken only at these points because the body parts are not self-occluding at these points, and this is a repeatable point in the walk action to record similar measurements.

3.4 Walking-Space Adjustment

The static body parameters recovered from subjects, from a single view angle, produce high discrimination power. When comparing across views, however, discrimination power decreases. The most obvious reason is that forshortening changes the value of many of the features. Furthermore, variations in how the part labeling techniques work in the different views can lead to a systematic variation between the views. And finally, other random error can occur when doing vision processing on actual imagery; this error will tend to be larger across different views.

In this paper we did not attempt to adjust for random error, but instead compensate for a variety of systematic error including forshortening. We assume that the same systematic error is being made for all subjects for each view angle. Therefore, we can use one subject as a reference subject and use his vision data, from different view angles, to find a scale factor to convert his vision data to a common frame using his motion-capture data as the reference.

Motion-capture data of a reference subject, is considered to be the ground truth information from the subject with minimal error. Our motion-capture system uses magnetic sensors to capture the three-dimensional position and orientation of the limbs of the subject as he (or she) walks along a platform. Sixteen

sensors in all are used: (1) head, (2) torso, (1) pelvis, (2) hands, (2) forearms, (2) upper-arms, (2) thighs, (2) calfs, (2) feet. If the error is truly systematic, then the scale factor found, using the motion-capture system, can be applied to the other subjects' vision data.

To achieve this, we model the error as a simple scaling in each dimension of the 4D-walk vector, which can be removed by a simple constant scale factor for each dimension. A mean 4D-walk vector

$$\bar{\mathbf{x}} = \langle d_{x1}, d_{x2}, d_{x3}, d_{x4} \rangle$$

from motion-capture walking sequences of a reference subject is recovered. Next, several (vision recovered) 4D-walk vectors,

$$\mathbf{w}_{ij} = \langle d_{w1}, d_{w2}, d_{w3}, d_{w4} \rangle$$

where i is the view angle and j is the walk vector number, are found of the reference subject from the angle-view, the near-side-view, and the far-side-view. The walk-vector, $\bar{\mathbf{x}}$, from the motion-capture system is used to find the constant scale factors needed to convert the vision data of the reference subject for each dimension and view angle separately by:

$$\mathbf{S}_{ij} = \langle \frac{d_{x1}}{d_{w1}}, \frac{d_{x2}}{d_{w2}}, \frac{d_{x3}}{d_{w3}}, \frac{d_{x4}}{d_{w4}} \rangle$$

where \mathbf{S}_{ij} is scale factor vector for view angle i and walk vector j , and the scale factor vector for a given view angle is

$$\mathbf{SF}_i = \langle sf_1, sf_2, sf_3, sf_4 \rangle = \frac{1}{N} \sum_{j=1}^N S_{ij}. \quad (7)$$

The 4D-walk vectors of each subject are converted to walking-space by

$$\mathbf{w}_{ij} \cdot \mathbf{SF}_i = \langle d_1 \cdot sf_1, d_2 \cdot sf_2, d_3 \cdot sf_3, d_4 \cdot sf_4 \rangle.$$

3.5 Results

We recorded 18 subjects, walking at the angle-view, far-side-view, and near-side-view. There are six data points (walk vectors) per subject for the angle-view, three data points per subject for the side-view far away, and three data per subject for the side-view close up yielding 108 walk vectors for the angle-view and 108 walk vectors for the side-view (54 far way, and 54 close up). The results are listed in Table II.

Table II is divided into two sets of results: Expected Confusion and Recognition Rates. The Expected Confusion is the metric discussed in Section 2. The Recognition Rates are obtain using Maxim Likelihood. Where, recognition is computed by modeling each individual as a single Gaussian and selecting the class with the greater likelihood.

Results are reported from the angle-view, near-side-view and far-side-view. Finally results are posted after the vision data was scaled to walking-space using

Table 1. The results of the multi-view gait-recognition method using static body parameters.

<i>Viewing Condition</i>	<i>Expected Confusion</i>	<i>Recognition Rates</i>
Angle View	1.53%	100%
Side View Far	.71%	91%
Side View Near	.43%	96%
Side View Adjusted (far and near)	4.57%	100%
Combine Angle and Side Views Adjusted	6.37%	94%

the appropriate scale factor based on the viewing condition. The results in the last row, titled *Combine Angle and Side Views Adjusted*, are the numbers of interests because this data set contains all data adjusted using the walking-space adjustment technique.

Once the data points are adjusted by the appropriate scale factors the expected confusion of the Side View (combining near and far) is only 4.57%. Also, the Combined Angle and Side views yield an expected confusion of 6.37%. This tells us that an individual's static body parameters will yield on average 6% confusion with another individual's parameters under these different views.

4 Conclusion

This paper has demonstrated that gait recognition can be achieved by static body parameters. In addition, a method to reduce the variance between static body parameters recovered from different views was present by using the actual ground truth information (using motion-capture data) of the static body parameters found for a reference subject. As with any new work, there are several next steps to be undertaken. We must expand our database to test how well the expected confusion metric predicts performance over larger databases. Experiments must be ran under more view angles, so the error over other possible views can be characterize. Also, the relationship between the motion-capture data and vision data needs to be explored further to find the best possible scaling parameters to reduce the expected confusion even lower than presented here. Lastly, in this paper we compensated for systematic error, and not random error. In future work, we will analyze how to determine random error, and attempt to compensate for (or minimize the effects of) the random error.

References

1. Bobick, A. F. and A. Y. Johnson, "Expected Confusion as a Method of Evaluating Recognition Techniques," Technical Report GIT.GVU-01-01, Georgia Institute of Technology, 2001. <http://www.gvu.gatech.edu/reports/2001/>.
2. Cover, T. M. and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, Inc., New York, 1991.
3. Cunado, D., M. S. Nixon, and J. N. Carter, "Automatic Gait Recognition via Model-Based Evidence Gathering," accepted for *IEEE AutoID99*, Summit NJ, 1999.
4. Cutting, J. and L. Kozlowski, "Recognizing friends by their walk: Gait perception without familiarity cues," *Bulletin of the Psychonomic Society* **9** pp. 353–356, 1977.
5. Davis, J.W. and A.F. Bobick, "The representation and recognition of action using temporal templates," *Proc. IEEE Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, pp 928–934, 1997.
6. Haritaoglu, I., D. Harwood, and L. Davis, "W4: Who, When, Where, What: A real time system for detecting and tracking people," *Proc. of Third Face and Gesture Recognition Conference*, pp. 222–227, April 1998.
7. Huang, P.S., C. J. Harris, and M. S. Nixon, "Human Gait Recognition in Canonical Space using Temporal Templates," *IEEE Procs. Vision Image and Signal Processing*, **146**(2), pp. 93–100, 1999.
8. Kozlowski, L. and J. Cutting, "Recognizing the sex of a walker from a dynamic point-light display," *Perception and Psychophysics*, **21** pp. 575–580, 1977.
9. Little, J.J. and J.E. Boyd, "Recognizing people by their gait: the shape of motion," *Videre*, **1**, 1996.
10. Murase, H. and R. Sakai, "Moving object recognition in eigenspace representation: gait analysis and lip reading," *Pattern Recognition Letters*, **17**, pp. 155–162, 1996.
11. Niyogi, S. and E. Adelson, "Analyzing and Recognizing Walking Figures in XYT," *Proc. Computer Vision and Pattern Recognition*, pp. 469–474, Seattle, 1994.
12. Stevenage, S., M. S. Nixon, and K. Vince, "Visual Analysis of Gait as a Cue to Identity," *Applied Cognitive Psychology*, **13**, pp. 00-00, 1999.

New Area Based Metrics for Gait Recognition

J.P. Foster, M.S. Nixon, and A. Prugel-Bennett

University of Southampton, Southampton, SO17 1BJ, UK
{jpf99r,msn,apb}@ecs.soton.ac.uk

Abstract. Gait is a new biometric aimed to recognise a subject by the manner in which they walk. Gait has several advantages over other biometrics, most notably that it is a non-invasive and perceivable at a distance when other biometrics are obscured. We present a new area based metric, called gait masks, which provides statistical data intimately related to the gait of the subject. Early results show promising results with a recognition rate of 90% on a small database of human subjects. In addition to this, we show how gait masks can also be used on subjects other than humans to provide information about the gait cycle of the subject.

Introduction

Gait is a new biometric primarily aimed at recognising a subject by the way they walk. Gait has several advantages over other biometrics. It is difficult to disguise (in fact disguising ones gait only has the effect of making oneself look more suspicious!) and gait can be recognised from a large distance where other biometrics fail and it is a non-invasive technique.

Medical work from Murray [1,2] supports the view that if all gait movements are considered then gait is unique. Psychological research from Johansson [3] shows that humans have a remarkable ability to recognise different types of motion. Johansson performed an experiment with moving light displays attached to body parts and showed that human observers can almost instantly recognise biological motion patterns, even when presented with only a few of these moving dots.

A more recent study by Stevenage [4] again confirmed the possibility of recognising people by their gait, but now using video. The study confirmed that, even under adverse conditions, gait could still be perceived. Psychological and medical studies therefore clearly support the view that gaits can be used for recognition.

Overview

This report will briefly look at previous approaches to gait recognition and present a new area based metric for gathering statistical data intimately related to the nature of gait. The technique will be performed upon both human and animal silhouettes and differences between the two will be examined.

Figure 1 shows an example of silhouette extraction from a human gait sequence. In this report we assume that the subject is walking normal to the camera's plane of view. The silhouettes are extracted using a combination of background subtraction and thresholding.

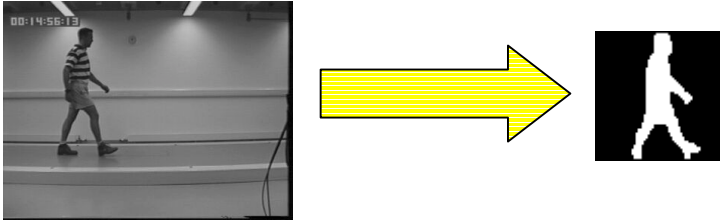


Fig. 1. Single Image from Gait Sequence converted to a silhouette.

Previous Approaches to Gait Recognition

Approaches to gait masks can be divided into two categories, model based and holistic. Holistic approaches aim to derive data from the human walking sequence that is similar for each subject but different for different people. Examples of this type of approach include Murase and Sakai [5], Huang [6] and Little and Boyd [7]. Model based approaches aim to explicitly model human motion and rely on human movement being tracked and a model being fitted to the image data. An example of this approach is by Cunado [8]. Performance of this technique was good with high recognition rates, however the computational costs of this approach are high.

Gait Masks Using Human Silhouettes

Gait masks are a new approach to gait recognition aiming to combine some of the elements of both the holistic and model based approaches. The disadvantage of traditional holistic approaches is that they simply derive data that is different for each class. They have no knowledge of what each class represents; given pictures of an elephant rather than a human subject and a traditional holistic approach would try to classify the subject in the same way. In contrast, model based approaches directly recognise gait by using equations, however this would be difficult to extend to non-human gait.

Gait masks aim to combine holistic and model-based approaches by using statistical data that is intimately related to the nature of gait. Gait masks are used to transform human silhouettes into information directly related to the gait of the subject.

Figure 2 shows an example of some gait masks.



Fig. 2. Sample gait masks.

Each gait mask aims to isolate a portion of the image and measure the area change within that area. The gait masks were chosen intuitively to represent area of the image likely to provide meaningful data about the gait of a subject. **Figure 3** shows examples of the results from Figure 2 on human silhouettes.

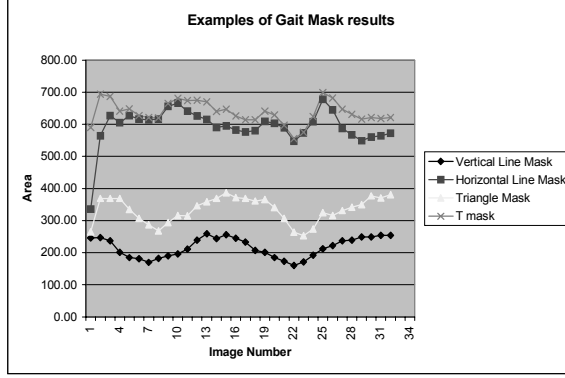


Fig. 3. Sample Gait Mask Results from one subject using 4 separate gait masks.

The gait masks are combined with the silhouettes using the procedure detailed below. The set C_p labels each set of sequences of each individual person). Each item in the set (S_{p_j}) represents sequence j of person p .

$$C_p = \{S_{p_j}\} \quad (1)$$

The set T_{p_j} represents the individual images (silhouettes) in each sequence from each subject. Each member of the set $S_{p_j}(t)$ represents a specific image (t) as a vector from person p sample j .

$$T_{p_j} = \{S_{p_j}(t)\} \quad (2)$$

The set K represents the set of gait masks, M_n , where each member represents each individual gait mask, each represented as a vector.

$$K = \{M_n\} \quad (3)$$

For each gait mask, n , and each person p , and each sample j , \mathbf{R} is the projection of each sample into a new space using gait mask \mathbf{M}_n .

$$\mathbf{R}_{np_j}(t) = \mathbf{M}_n \cdot \mathbf{S}_{p_j}(t) \quad (4)$$

The vertical line mask produces output that is sinusoidal in nature. The peaks in the graph represent when the legs are closest together and the dips represent where the legs are at furthest flexion. The gait masks are therefore providing statistics that are intimately related to the mechanics of gait.

It is possible to use these graphs, produced using the gait masks, to provide recognition capabilities. By comparing the graphs using least squares and using a database of all samples and finding the closest match, recognition rates of over 80% are possible (dependent on the mask chosen). Since the subject may start his walking cycle at a different point, the graphs are compared at all possible shifts along the axis and the maximal correlation is taken. **Table 1** shows the recognition rates of various gait masks using this technique.

Table 1. Recognition results from various gait masks.

Gait Mask	Recognition Rate
Horizontal Line Mask	71%
Vertical Line Mask	83%
Top Half Mask	58%
Right Half Mask	42%
Left Half Mask	33%
Bottom Left Mask	42%
Bottom Right Mask	38%
Triangle Mask	63%
T Mask	71%
Bottom Half Mask	54%
Mid Triangle Mask	83%

The results from the vertical line mask (recognition rate 83% on a database of six subjects with four samples each) were most encouraging as the output from this mask was directly related to the gait of the subject. However, once the input silhouettes were corrupted by even small amounts of noise the recognition rate dropped dramatically. Consequently, we decided to look at the results from this mask in greater detail.

To recognise a subject by their gait we are primarily concerned with the AC component of the data, that is the temporal component of the data. The DC component of the data represents static data about the subject; this could easily change due to the subject carrying something for example. The temporal nature of gait should be consistent amongst samples of the same person.

Figure 4 shows the sinusoidal patterns generated using the vertical line mask from 4 different people. Using the least square correlation technique, poor recognition rates were achieved. Using a more sophisticated technique, Canonical Analysis (CA), resulted in a dramatic performance increase.

CA was used to produce a set of axes on which to project the data (the canonical axes). The data was divided into a set of training and test data. The training data consisted of three samples from each of the six subjects. The centroid of each subject in canonical space was recorded and the distance between this centroid and the test data was also noted. This was then used to calculate the recognition rate on the SOTON database (which consists of 6 subjects with 4 samples each). Initial results were promising with a recognition rate of over 80% and good class separability.

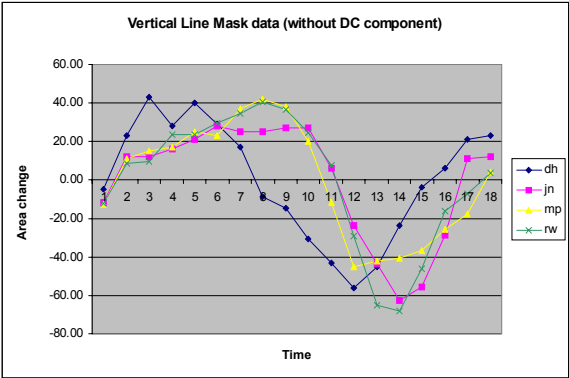


Fig. 4. Sinusoidal Patterns from 4 people using Vertical Line Mask.

To further evaluate the performance of the new technique the system was also tested on a larger database consisting of the SOTON database and each of the samples corrupted with various amounts of noise (1%, 2%, 4% and 8% noise). Performance remained high with a recognition rate of over 80% even in the presence of noise.

Gait Masks Using Animal Data

Gait masks can be used to quickly distinguish between the motion of a biped (e.g. a human) and a quadruped. The single vertical line mask was used to provide data to discriminate between human subjects. Using this technique with animal data yields ineffectual results that provide no information about the gait of the subject. This is simply because the center of a quadruped provides very little temporal information. **Figure 5** illustrates this.

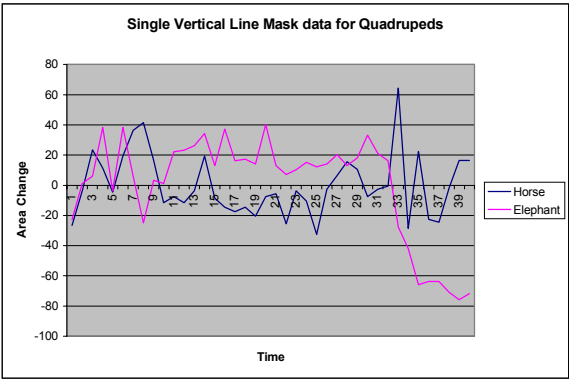


Fig. 5. Vertical Line Mask data using Animal Silhouettes.

To provide information more relevant to the subject being analysed the gait masks were modified. Two vertical line masks were used, instead of the single vertical line mask used for human gait. By using these new gait masks it is possible to extract information relative to the subject being studied. **Figure 6** illustrates this:

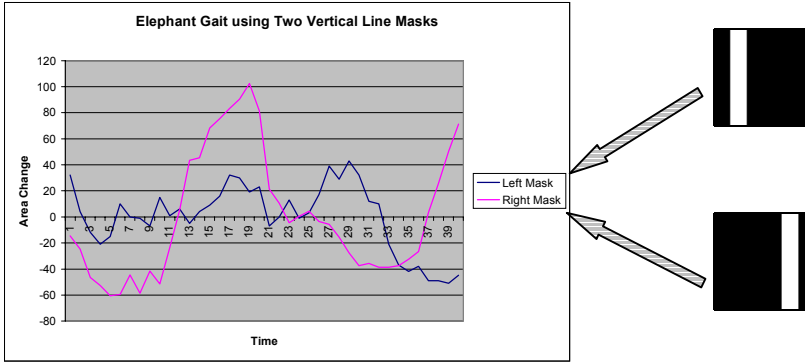


Fig. 6. Gait of an elephant described by Two Vertical Line Masks.

By using the right mask the sinusoidal pattern of motion is clearly evident. The left mask data is not of a sinusoidal nature which is due to the trunk of the elephant swinging across the legs. This shows how the gait masks need to be adapted for each animal analysed. It also illustrates how gait masks can be used to provide information about gait for subjects other than humans.

Conclusions

We have presented a new area based metric for gait recognition that produces good results on a small database. We have shown (by removing the DC component of the data) that recognition is possible by only using the temporal components of the silhouette sequence. The technique produces encouraging results on a database of human subjects with recognition rates of over 80% even in the presence of noise.

Additionally we have shown the basic premise of gait masks is applicable to areas other than using human gait as a metric. Gait masks can provide information about the walking cycle that could be used to provide information such as the cadence of the subject.

References

- [1] M.P. Murray, Gait as a total pattern of movement , *American Journal of Physical Medicine*, **46**, no. 1, pp. 290-332, 1967.
- [2] M.P. Murray, A.B. Drought, and R.C. Kory, Walking patterns of normal men , *Journal of Bone Joint Surgery*, **46-A**, no. 2, pp. 335-360, 1964.
- [3] G. Johansson, Visual perception of biological motion and a model for its analysis , *Perception Psychophysics*, **14 (2)**, pp. 201-211, 1973.
- [4] S.V. Stevenage, M.S. Nixon, and K. Vince, Visual Analysis of Gait as a Cue to Identity , *Applied Cognitive Psychology*, **13** p. 513-526, 1999.
- [5] H. Murase and R. Sakai, Moving object recognition in eigenspace representation: gait analysis and lip reading , *Pattern Recognition Letters*, **17**, pp. 155-162, 1996.
- [6] P.S. Huang, C.J. Harris, and M.S. Nixon, Human Gait Recognition in Canonical Space using Temporal Templates , *IEEE Proc. VISP*, **146(2)**, pp. 93-100, 1999.
- [7] J. Little and J. Boyd, Describing motion for recognition , *Videre*, **1(2)**, pp. 1-32, 1998.
- [8] D. Cunado, M.S. Nixon, and J.N. Carter, Automatic Gait Recognition via Model-Based Evidence Gathering , *Proc. AutoID99: IEEE Workshop on Automated ID Technologies*, Summit, pp. 27-30, 1999.

On-Line Signature Verifier Incorporating Pen Position, Pen Pressure, and Pen Inclination Trajectories

H. Morita, D. Sakamoto, T. Ohishi, Y. Komiya, and T. Matsumoto

Department of Electrical, Electronics, and Computer Engineering
Waseda University 3-4-1 Ohkubo, Shinjuku-ku, Tokyo, Japan, 169-8555
takashi@mse.waseda.ac.jp

Abstract. This paper proposes a new algorithm PPI (pen-position/pen-pressure/pen-inclination) for on-line pen input signature verification. The algorithm considers writer's signature as a trajectory of pen-position, pen-pressure and pen-inclination which evolves over time, so that it is dynamic and biometric. Since the algorithm uses pen-trajectory information, it naturally needs to incorporate stroke number (number of pen-ups/pen-downs) variations as well as shape variations. The proposed scheme first generates templates from several authentic signatures of individuals. In the verification phase, the scheme computes a distance between the template and input trajectory. Care needs to be taken in computing the distance function because; (i) length of a pen input trajectory may be different from that of template even if the signature is genuine; (ii) number of strokes of a pen input trajectory may be different from that of template, i.e., the number of pen-ups/pen-downs obtained may differ from that of template even for an authentic signature. If the computed distance does not exceed a threshold value, the input signature is predicted to be genuine, otherwise it is predicted to be forgery. A preliminary experiment is performed on a database consisting of 293 genuine writings and 540 forgery writings, from 8 individuals. Average correct verification rate was 97.6 % whereas average forgery rejection rate was 98.7 %. Since no fine tuning was done, this preliminary result looks very promising.

1 Introduction

Authentication of individuals is rapidly becoming an important issue. This paper proposes a new biometric authentication scheme using online signature trajectories, and reports a preliminary experimental result. The scheme is different from the related works [1]-[5].

2 The Algorithm

2.1 Feature Extraction

The raw data available from our tablet (WACOM Art Pad 2 pro Serial) consists of five dimensional time series data (Fig 2.1):

$$(x(t_i), y(t_i), p(t_i), px(t_i), py(t_i)) \in R^2 \times \{0,1,...,255\} \times R^2 \quad (2.1) \\ i = 1,2,...,I$$

where $(x(t_i), y(t_i)) \in R$ is the pen position at time t_i , $p(t_i) \in \{0, 1, \dots, 255\}$ represents the pen pressure, $px(t_i)$ and $py(t_i)$ are pen inclinations with respect to the x - and y -axis (This tablet automatically returns the inclinations.), $t_i - t_{i-1} \approx 5ms$ so that there are too many points which is not appropriate. Uniform resampling often results in a loss of important features. Consider, for instance, the raw data given in Fig 2.2(a). If one resamples the data uniformly then the sharp corner may be lost as is shown in Fig 2.2(b). Our resampling procedure checks if

$$\theta_i := \tan^{-1} \frac{y(t_i) - y(t_{i-1})}{x(t_i) - x(t_{i-1})} \leq \theta^* \quad (2.2)$$

where θ^* is a threshold value. If (2.2) holds, then $((x(t_i), y(t_i)))$ is eliminated, otherwise it is kept. This typically gives Fig 2.2(c) which retains a sharp corner while portions of pen trajectory without sharp corners retain information with smaller number of points. This is a preprocessing done in our pen-input on-line character recognizers which worked very well [6][7]. Details are omitted. Let

$$\Delta f_i := \sqrt{(x(t_i) - x(t_{i-1}))^2 + (y(t_i) - y(t_{i-1}))^2} \quad (2.3)$$

then our feature consists of the following five dimensional data

$$(\theta_j, \Delta f_j, p_j, px(t_i), py(t_i)) \in R^2 \times \{0, 1, \dots, N\} \times R^2 \quad (2.4)$$

$$i = 1, 2, \dots, I \quad j = 1, 2, \dots, J$$

Our verification algorithm described below computes a weighted sum of three different distance measures between an input data and stored templates.

2.2 Distance Measure

Let

$$(\eta_l, \Delta g_l, q_l, qx(t_k), qy(t_k)) \in R^2 \times \{0, 1, \dots, N\} \times R^2 \quad (2.5)$$

$$k = 1, 2, \dots, K \quad l = 1, 2, \dots, L$$

be the resampled feature trajectory of a template and consider

$$|\theta_j - \eta_l| d(p_j, q_l) \rho(\Delta f_j, \Delta g_l) \quad (2.6)$$

where $d(p_i, q_i) := |p_i - q_i| + 1$ incorporates pen-pressure information. The last term "1" is to avoid zero value of a $d(p_i, q_i)$. Function ρ is defined by

$$\rho(\Delta f_j, \Delta g_l) := \sqrt{\Delta f_j^2 + \Delta g_l^2} \quad (2.7)$$

which is to take into account local arc length of the trajectories. Generally $K \neq I, L \neq J$ even when signatures are written by the same person so that time-warping is necessary to compute (2.6) over the whole trajectories. The following is our angle arc length distance measure

$$D1 := \min_{\substack{j_s \leq j_{s+1} \leq j_s+1 \\ l_s \leq l_{s+1} \leq l_s+1}} \sum_{s=1}^S |\theta_{j_s} - \eta_{l_s}| d(p_{j_s}, q_{l_s}) \rho(\Delta f_{j_s}, \Delta g_{l_s}) \quad (2.8)$$

where $j_1 = l_1 = 1, j_s = J, l_s = L$ are fixed. The first factor in (2.8) measures the difference of the directions of the two pen trajectories, the second factor puts weight according to the pen-up/pen-down information of the input and the template, and the last term puts penalty on the input trajectory length which deviates from that of the template trajectory. Because of the sequential nature of the distance function, Dynamic Programming is a feasible means of the computation:

$$D1(0,0) = 0$$

$$D1(j_s, l_s) = \min \begin{cases} D1(j_s - 1, l_s - 1) + |\theta_{j_s} - \eta_{l_s}| \times d(p_{j_s}, q_{l_s}) \rho(\Delta f_{j_s}, \Delta g_{l_s}) \\ D1(j_s - 1, l_s) + |\theta_{j_s} - \eta_{l_s}| \times d(p_{j_s}, q_{l_s}) \rho(\Delta f_{j_s}, 0) \\ D1(j_s, l_s - 1) + |\theta_{j_s} - \eta_{l_s}| \times d(p_{j_s}, q_{l_s}) \rho(0, \Delta g_{l_s}) \end{cases}$$

$$d(p, q) = |p - q| + 1$$

$$\rho(\Delta f, \Delta g) = \sqrt{\Delta f^2 + \Delta g^2}$$

$$D2 := \min_{\substack{i_s' \leq i_{s'+1} \leq i_s'+1 \\ k_s' \leq k_{s'+1} \leq k_s'+1}} \sum_{s'=1}^{S'} |px_{i_s'} - qx_{k_s'}|, \quad D3 := \min_{\substack{i_s'' \leq i_{s''+1} \leq i_s''+1 \\ k_s'' \leq k_{s''+1} \leq k_s''+1}} \sum_{s''=1}^{S''} |py_{i_s''} - qy_{k_s''}| \quad (2.9)$$

Pen-inclination distances are defined as (2.9). Those are computable via DP also.

Figure 2.4(a) is a scatter plot of $(D1, D2, D3)$ consisting of 150 authentic signatures (triangle) and 351 forgery signatures (square) taken from eight individuals. Figure 2.4(b), (c), and (d) show the projections onto the $(D1, D2)$ -plane, $(D2, D3)$ -plane and $(D1, D3)$ -plane respectively.

These plots naturally suggest that there should be a two dimensional surface which could separate authentic signatures from forgeries reasonably well even though perfect separation may not be achieved.

In order to explain our template generation procedure, recall two types of errors in signature verification: a) Type I Error (False Rejection Error), b) Type II Error (False Acceptance Error)

Given m_0 authentic signature trajectories, divide them into two group S_1 and S_2 consisting of m_1 and m_2 trajectories, respectively, where the former is to generate templates while the latter is for verification test. We compute the total squared

distance $D^2 = (D1)^2 + (D2)^2 + (D3)^2$ between each of the signatures in S_1 and sort them according to their distances between each other. Choose three signatures with the smallest D^2 . These three will be used as templates.

In order to select the threshold value for distance between input and template, compute the $3 \times (m_1 - 3)$ distances between the chosen three and the remaining $m_1 - 3$ signatures and let the threshold value Th be the average of five largest distances.

Note that three template signatures are generated for each individual. Given an input signature, compute the squared distance measure between it and the three templates and let $(D_{\min})^2$ be the smallest. We introduce a parameter $c \in [0.5, 2.0]$ to be selected and the input is predicted to be authentic if $(D_{\min})^2 \leq c \cdot Th$ while the input is predicted as a forgery if $(D_{\min})^2 > c \cdot Th$.

3 Experiment

This section reports our preliminary experiment using the algorithm described above. Eight individuals participated the experiment. The data were taken for the period of three months. There are 861 authentic signatures, 1921 forgery signatures and 205 signatures for template generation. Table 3.1 shows the details. Figure 3.2 shows average verification error as a function of parameter c described above, where the intersection between Type I Error and Type II Error curves gives 3.0%. Figure 3.3 shows the error curves of individual "B" where zero error is achieved at $c = 1.1$.

Figure 3.1(a) is an unsuccessful attempt of a forgery rejected by our algorithm while Fig. 3.1(b) is an authentic signature accepted by the PPI.

Experiments with larger number of users will be needed.

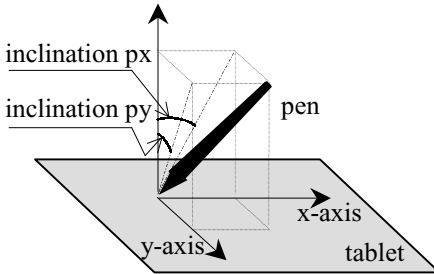


Fig. 2.1 Raw data from tablet .

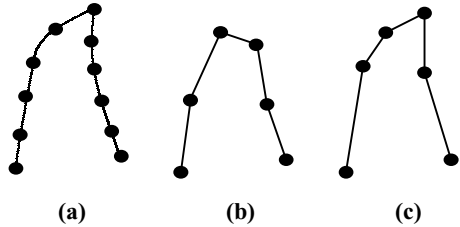


Fig. 2.2 Our resampling algorithm preserves sharp corners.

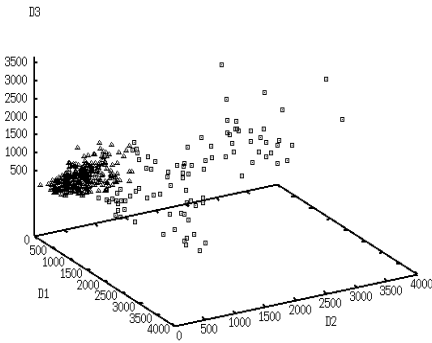
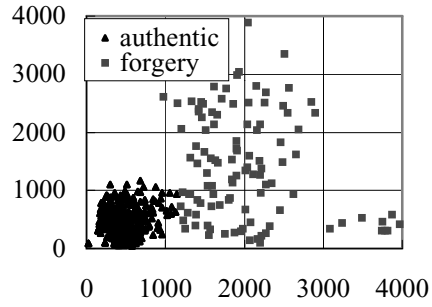
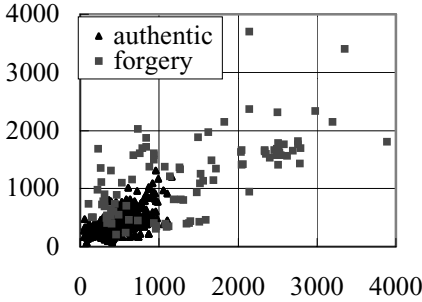


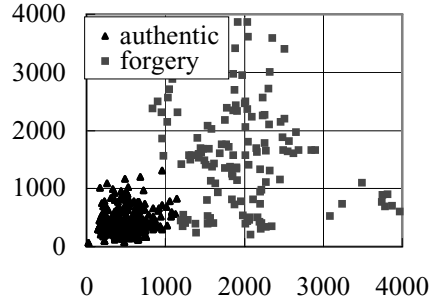
Fig. 2.4(a) Scatter plot of $(D1, D2, D3)$.



(b) Projection onto the $(D1, D2)$ -plane.



(c) Projection onto the $(D2, D3)$ -plane.



(d) Projection onto the $(D1, D3)$ -plane.

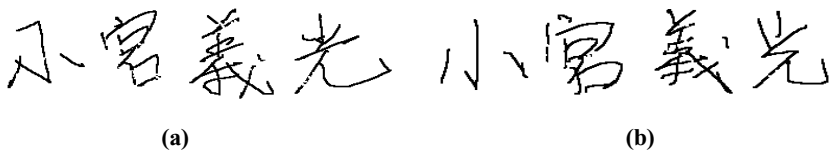
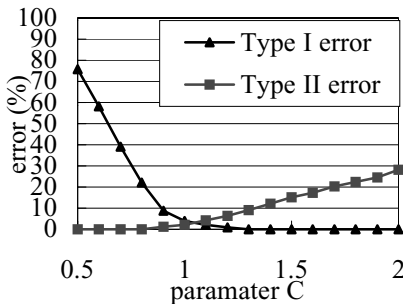
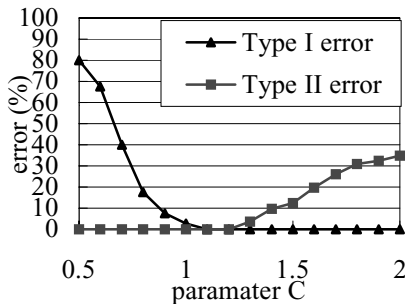


Fig. 3.1(a),(b) Forgery rejected by the PPI algorithm and Genuine signature accepted by the PPI.

Table 3.1 Data for Experiment.

Individuals	authentic		forgery	total
	test	Template generation	test	
A	184	45	585	814
B	40	10	81	131
C	126	30	237	393
D	24	6	68	98
E	173	39	435	473
F	52	12	71	135
G	172	42	288	502
H	91	21	156	268
total	861	205	1921	2987

**Fig. 3.2** Average verification error.**Fig. 3.3** The error curves of individual B .

References

1. T. Tabuki, Personal Identification by Signature, Proc. Seminar on Sensing Technology and Applications, Osaka, Japan, 1997.
2. M. Kato and Y. Kawashima, Signature Verification Using Online Data such as Pen Pressure and Velocity , IPSJ, pp. 2-199, 1993.
3. H. Taguchi, et al. On-Line Recognition of Handwritten Signatures by Feature Extraction of the Pen Movement ,IEICE, D Vol.J71-D No.5 pp. 830-840, 1988.
4. Chang Ji Jin et al. On-Line Signature Verification by Non-public Parameters , IEICE, D-II Vol.J75-D-II No.1 pp. 121-127, 1992.
5. I Yoshimura and M Yoshimura, On-line Signature Verification Incorporating the Direction of Pen Movement , Trans.IEICE Jpn., E74, pp. 2803-2092, 1991.
6. S.Masaki et al. An On-Line Hand-writing Character Recognition Algorithm RAV (Re-parameterized Angle Variations) , 4th ICDAR, Proceedings Vol.2, pp. 919-925, 1997.
7. T.Takahashi et al. On-LineHandwriting Character Recognition Using Hidden Markov Models , 4th ICDAR97, Proceed-ings Vol.1, pp. 369-375, 1997.
8. Y. Komiya and T. Matsumoto. "On-line Signature Verification using Position, Pressure, Pen Angles", Proc. National Convention, IEICE, D-12-44, pp. 217, 1999.

Iris Recognition with Low Template Size

Raul Sanchez-Reillo¹ and Carmen Sanchez-Avila²

¹ Carlos III University of Madrid, Dpt. Electric, Electronic and Automatic Engineering
c/Butarque, 15, E-28911 Leganes, Madrid, Spain
rsreillo@ing.uc3m.es

² Polytechnic University of Madrid, E.T.S.I. Telecomunicacion, Dpt. Applied Mathematics
Ciudad Universitaria, s/n, E-28040 Madrid, Spain
csa@mat.upm.es

Abstract. Among all the biometric techniques known nowadays, Iris Recognition is taken as the most promising of all, due to its low error rates without being invasive and with low relation to police records. Based on Daugman's work, the authors have developed their own Iris Recognition system, obtaining results that show the performance of the prototype and proves the excellences of the system initially developed by Daugman. A full coverage of the pre-processing and feature extraction blocks is given. Special efforts have been applied in order to obtain low template sizes and fast verification algorithms. This effort is intended to enable a human authentication in small embedded systems, such as an Integrated Circuit Card (smart cards). The final results show viability of this target, enabling a template size down to 256 bits. Future works will be focussed in new feature extraction algorithms, as well as optimising the pre-processing block.

1 Introduction

User authentication plays a very important role in the security of an application. Nowadays, this authentication is performed via passwords, Personal Identification Numbers (PINs) or Identification Tokens (such as smart cards). Unfortunately, those means of authentication cannot provide a high level of security because they can be copied, inspected and/or stolen. They only show the knowledge of some data or belonging of a determined object, not a real authentication of the user. Biometrics is the only way to identify a person with sufficient legal background. From all the techniques that exist nowadays (voice, fingerprint, hand geometry, etc.) iris recognition is the most promising for the environments mentioned [3].

The potential of the human iris for biometric identification comes from the anatomy of the eye [1]. The iris is a dynamical tissue that is highly protected against the outer by the cornea and whose modification implies surgery with a high risk of damaging the vision. As it reacts to changes in illumination, the detection of a dead or plastic iris is very easy, avoiding this kind of counterfeit. Also several studies have shown that normal variations in colouring and architecture of the tissues of the iris are so multitudinous that there are not ever two irises alike, not even for uniovular (identical) twins [3]. Even for a single person his two irises are also different. This

leads to the fact that in an iris recognition system, the False Acceptance Rate (FAR), i.e. the probability of an intruder entering the system [4], is null. And through good processing algorithms and optimal selection of the template for each user, a False Rejection Rate (FRR), i.e. the probability of an authorised user to be rejected by the system, could be below 5% for a single try system.

A biometric system is based on four blocks, where the first one is in charge of capturing the biological data of the user. The data should be pre-processed in order to be adapted to the necessities of the feature extraction block. The features extracted are then verified against a template previously stored. If the verification is successful, the access is granted, denying it in other case.

The user's template is obtained in the enrolment process, where a photo of the user's eye is taken, pre-processed, and its features extracted. Those features are stored for comparison with each user's sample.

The authors have developed their own Iris Recognition system, based on the work being carried by Daugman [3] (who is considered the father of this technique). Their development of each of the blocks is described. First, an explanation to the pre-processing of the human iris is presented, following a small description of the way the data is captured. Then the feature extraction block is shown and the verification algorithm used. Results, changing the template size, are given and conclusions obtained.

2 Capture and Pre-processing

Being the cornea transparent, the users' samples can be obtained throughout a high-resolution photograph, or a video sequence. Special optics are applied to have a zoom that enables the capture of the image of the iris from a distance large enough to avoid any user rejection. Here, a photograph is taken, covering the whole eye of the user (Fig. 1.a).

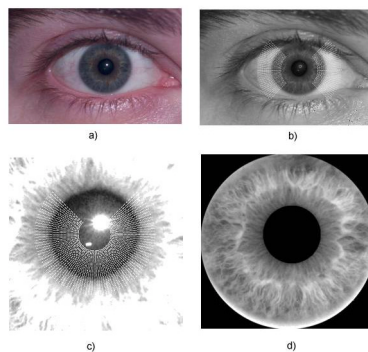


Fig. 1. Iris pre-processing: a) Photograph taken; b) Sclera and c) Pupil detection; d) Result.

One important characteristic of the system is the fact that the eyelids almost always cover some part of the iris (the superior or inferior cone). Therefore, during the whole process, the algorithms will be based on its lateral cones [6],[7].

The pre-processing block starts converting the image to greyscale and stretching its histogram. Then, throughout a gridding process, the centre of the iris, as well as the outer boundary, i.e. the border between the iris and the sclera, is detected taking profit of the circular structure of the iris. The detection is performed maximising D in the equations shown in the following equation, where (x_0, y_0) is a point in the grid taken as centre, Δ_r, Δ_θ are the increments of radio and angle, and $I(x, y)$ is the image in grey levels.

$$D = \sum_m \sum_{k=1}^5 (I_{n,m} - I_{n-k,m}) \quad (1)$$

$$I_{i,j} = I(x_0 + i\Delta_r \cos(j\Delta_\theta), y_0 + i\Delta_r \sin(j\Delta_\theta))$$

Once detected the outer bounds of the iris (Figure 1.b), everything in the image outside it is suppressed, and the same process is performed in order to find the inner boundary, i.e. the frontier between the iris and the pupil (Figure 1.c). The points inside this last border are also suppressed, obtaining the image shown in 1.d.

The last step in the pre-processing block is a transformation of the image. In this transformation the superior and inferior cones of the iris are eliminated, and the differences in the size of the iris are compensated throughout a polar sampling of the image, obtaining J as a result. The equation system used for this transformation is the following:

$$J(x, y) = IE(x_0 + r \cos \theta, y_0 + r \sin \theta)$$

$$r = r_i + (x - 1)\Delta_r, \quad \forall x \in N : x \leq \frac{r_e - r_i}{\Delta_r}$$

$$\theta = \begin{cases} -\frac{\pi}{4} + (y - 1)\Delta_\theta & , \text{if } y \leq \frac{\pi}{2\Delta_\theta} \\ \frac{3\pi}{4} + (y - 1)\Delta_\theta & , \text{if } y > \frac{\pi}{2\Delta_\theta} \end{cases}, \quad \forall y \in N : y \leq \frac{\pi}{\Delta_\theta} \quad (2)$$

where IE is the iris image with the sclera and pupil extracted, r_i and r_e are the inner and outer radii, (x_0, y_0) is the centre of the pupil, and Δ_r and Δ_θ are the sample intervals in magnitude and angle. Figure 2 shows this transformation graphically.

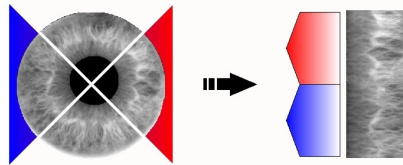


Fig. 2. Graphical representation of the transformation performed to the iris.

3 Feature Extraction and Verification

The image of the iris pre-processed, J , is weighted with the imaginary part of a Gabor filter [2], used in four orientations ($0, \pi/4, \pi/2$ and $3\pi/4$). In order to perform this operation, the image is divided in a determined number of sections (overlapped or not), and the following equation is applied [8]:

$$c(i, j) = \sum_{x=1}^N \sum_{y=1}^M J\left(i + x - \frac{N}{2}, j + y - \frac{M}{2}\right) \cdot g(x, y, \varphi_k, \lambda)$$

$$g(x, y, \varphi_k, \lambda) = \exp\left\{-\frac{1}{2}\left[\frac{(x \cos \varphi_k + y \sin \varphi_k)^2}{\sigma_x^2} + \frac{(-x \sin \varphi_k + y \cos \varphi_k)^2}{\sigma_y^2}\right]\right\} \cdot \sin\left\{\frac{2\pi(x \cos \varphi_k + y \sin \varphi_k)}{\lambda}\right\} \quad (3)$$

In these equations, the dimension of the filter is $N \times M$, (i, j) is the centre of each section and $\varphi_k, \lambda, \sigma_x$ and σ_y are the parameters of the filter, meaning orientation, scale and deviations in x and y respectively. The number of sections chosen will determine the size of the feature vector [5].

Verification is performed using a Hamming Distance:

$$d = \frac{1}{L} \sum_{i=1}^L x_i \oplus t_i \quad (4)$$

where L is the dimension of the feature vector and x_i and t_i are the i -th component of, respectively, the sample being verified, and the template. In order to enable the use of this distance, the features obtained ($c(i, j)$) are compared with a threshold giving a 0 if it is negative and 1 in any other case. Results of the Classification system can be seen in Figure 3a), while the ones obtained in the Verification system are shown in Figure 3b).

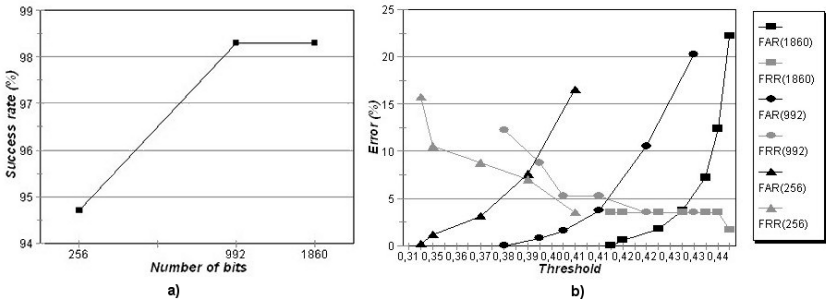


Fig. 3. Results obtained: a) in Classification; b) in Authentication.

These results have been obtained using a database of both eyes of 10 people and, at least, 10 photos of each eye. The photographs were taken in different hours and days, during 7 months. Each eye was considered as "from a different person", i.e., each person has two identities, the one of the left eye, and the one of the right one.

The results in classification show a classification rate of 98,3%, for 992 and 1860 bits, demonstrating the unicity of the iris features extracted. Also this proves that each iris of a single person have different characteristics, which leads to different features extracted.

In a verification system, the performance can be measured in terms of three different rates:

- False Acceptance Rate (FAR): the probability of identifying an intruder as an enrolled user.
- False Rejection Rate (FRR): the probability of rejecting an enrolled user, as if he were an intruder.
- Equal Error Rate (EER): the value where the FAR and FRR rates are equal.

It can be seen that, in all cases, the Equal Error Rate (EER), i.e. the cross point between the FAR and the FRR curves, is always below 10%, achieving a 3,6% for 1860 bits. But what is more important, is that a null FAR has been obtained for very low rates of False Rejection, which means that this system is optimal for very high security environments.

Further more, the FRR can be lowered, using the system as a 3-try-rejection system, i.e. rejecting a user after 3 consecutives errors (as it happens with a credit card in an Automatic Teller Machine).

4 Conclusions

A biometric identification technique based on the pattern of the human iris has been reported. This technique is well suited to be applied to any application needing a user authentication. The capture scheme and the pre-processing algorithms have been described, as well as the feature extraction block and the verification system. The mentioned high level of security has been shown with the error rates obtained in the author's prototype, achieving null False Acceptance without damaging the confidence of the user, due to its low FRR. Further efforts should be applied to reduce the high computational and economical cost involved in this system, changing the feature extraction algorithms and optimising the pre-processing block. Also video will be used instead of photograph images.

References

- [1] M.L. Berliner, "Biomicroscopy of the Eye". Paul B. Hoeber, Inc. 1949.
- [2] R. Carmona, W.L. Hwang, and B. Torrance, "Practical Time-Frequency Analysis", Vol. 9, Wavelet Analysis and its Applications. Academic Press, San Diego, 1998.
- [3] J.G. Daugman, "High Confidence Visual Recognition of Persons by a Test of Statistical Independence". IEEE Trans. PAMI, vol. 15, no. 11, Nov. 1993.
- [4] R.O. Duda and P.E. Hart, "Pattern Classification and Scene Analysis". John Wiley & Sons. 1973.
- [5] A.K. Jain, R. Bolle, S. Pankanti, et al., "Biometrics: Personal Identification in Networked Society". Kluwer Academic Publishers. EE.UU. 1999.
- [6] R. Sanchez-Reillo, C. Sanchez-Avila, and J.A. Martin-Pereda, "Minimal Template Size for Iris Recognition". Proc. of the First Joint BMES/EMBS International Conference (Atlanta, USA), 13-16th October, 1999. p. 972.
- [7] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos. "Multiresolution Analysis and Geometric Measure for Biometric Identification". Secure Networking - CQRE [Secure]99. Nov/Dec, 1999. Lecture Notes in Computer Science 1740, pp. 251-258.
- [8] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos. "Improving Access Control Security using Iris Identification". Proc. 34rd Annual 2000 International Carnahan Conference on Security Technology. Ottawa (Canada), 23-25 Oct, 2000. pp. 56-59.

RBF Neural Networks for Hand-Based Biometric Recognition

Raul Sanchez-Reillo¹ and Carmen Sanchez-Avila²

¹ Carlos III University of Madrid, Dpt. Electric, Electronic and Automatic Engineering
c/Butarque, 15, E-28911 Leganes, Madrid, Spain
rsreillo@ing.uc3m.es

² Polytechnic University of Madrid, E.T.S.I. Telecomunicacion, Dpt. Applied Mathematics
Ciudad Universitaria, s/n, E-28040 Madrid, Spain
csa@mat.upm.es

Abstract. A recognition system based on hand geometry biometrics is reported in this paper. The aim of this development is to improve the reliability of automatic identification systems. The capture of the data, as well as the pre-processing and feature extraction blocks is detailed. Once the features are obtained, they should enter the recognition block, which has been developed using Neural Networks. From the different Neural Networks existing nowadays, the Radial Basis Functions (RBF) ones have been chosen for their shorter training time, and the lack of randomness in the training algorithm. Results are analyzed as a function of the number of vectors of each user taken for the training of the net, obtaining up to 98,1% for only 5 samples of each user.

1 Introduction

In a security environment, one of the main tasks to be performed is the access control for determined areas. If this is needed to be made automatically, the typical solution adopted is the use of user authentication schemes based on passwords, secret codes and/or identification cards or tokens. Unfortunately, schemes based only on passwords or secret codes can be cracked by intercepting the presentation of such password, or even by counterfeiting it (via passwords dictionaries or, in some systems, via brutal force attacks). On the other hand, an intruder can attack systems based on identification card or tokens by robbing, copying or simulating that card or token. If the scheme used in the system is based both on a card and a password (usually called Personal Identification Number - PIN), the intruder should apply more effort to gain entry to the system, and with more advanced technologies, such as smart cards, some vulnerabilities of the system could be avoided (e.g. brutal force attacks are impossible under a well defined smart card).

Another way to solve the access control problem, is the use of biometrics, i.e. identifying a person through some biological and/or behavioral features. Biometrics are based on very different techniques such as speaker verification, signature recognition, or the measurement of fingerprint, iris pattern or hand geometry [1],[2].

Each biometric technique has its own advantages and disadvantages. While some of them provide more security, i.e. lower False Acceptance Rate (FAR) and False Rejection Rate (FRR), other techniques are cheaper or better accepted by final users.

The authors report here the work performed to obtain a biometric recognition system based on the geometrical pattern of the user's hand and Radial Basis Functions Neural Networks. Hand Geometry Measurement was chosen for being a medium/high security technique with a medium equipment cost (only a low resolution CCD camera is needed) and low computational cost (because the algorithms to extract the features are based on basic morphological image processing [3][4][5]) and very low feature vector size.

The scheme of a Biometric Recognition System using Neural Networks is shown in Fig. 1, where the two processes involved in the identification (enrollment and recognition) can be seen.

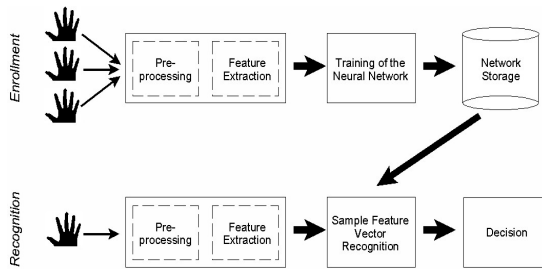


Fig. 1. Block diagram of a Biometric Recognition System.

During the enrollment, a set of samples from the users are taken, and processed until the features of each sample are extracted. Then, the neural network is trained, and when the training is complete, the parameters of the net are stored. When the user wants to gain access, a new sample is taken from him and new features are extracted. The feature vector is then used as the input in the previously stored network, and the output obtained is analyzed by a decision block which decides, based on a threshold, if the sample belongs to a user enrolled in the system or not.

Following the biometric scheme shown, this paper will be divided as follows. In section 2, all the processes needed to obtain the feature vector will be studied. Then, in section 3, all the facts concerning the recognition using RBF neural networks will be discussed, showing the results obtained. The last section will be a summary with the conclusions obtained.

2 Hand Geometry Biometrics

To obtain a feature vector of the user's hand, first an image of it should be obtained. This image is then pre-processed to be adapted to the special needs of the feature extraction block. This three processes are detailed in the following subsections.

2.1 Image Capture

An image of the hand is taken using a low resolution CCD camera. The hand is placed on a platform guided by 6 tops, in order to position the hand in front of the camera objective. Different views of the prototype designed can be seen in Fig. 2.

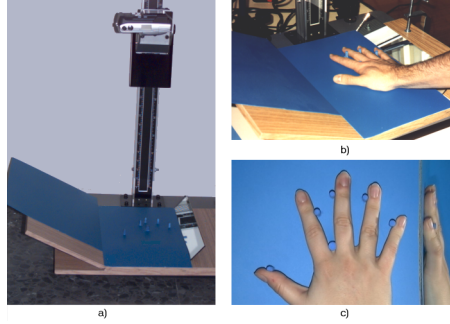


Fig. 2. Prototype designed: a) general view; b) detail of the platform and placement of the hand; c) photograph taken.

A mirror is located on one side of the platform for obtaining a side view of the hand and performing more measures. The platform is painted in blue to increase the contrast with the human skin.

2.2 Pre-processing

After the image is captured, it is processed to obtain the profile of the palm and of the side view of the hand. The colour components of the image are operated to obtain a high contrast image with the background suppressed. Then, a Sobel function is applied to detect the main edges of the image. An illustration of partial results obtained in the pre-processing block are shown in Fig. 3.

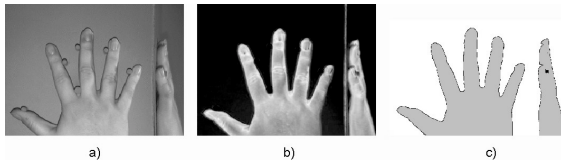


Fig. 3. Pre-processing of the hand: a) image captured; b) high contrast image; c) edges detected.

2.3 Features Extraction

From the edges detected, geometrical measures are performed to obtain a set of 25 features, including:

- Widths of the 4 fingers (the thumb is excluded).
- Width of the palm.
- Height of the palm, the middle finger and the small finger.
- Distances between the inter-finger points.
- Angles between the inter-finger points.

To avoid changes due to gain or losing of weight, all the measures (except the angles) are normalized by the first width measured of the middle finger. The result is a feature vector with 25 components that will be used as an input for the Neural Network, as it will be explained in the next section.

3 RBF-Based Recognition

Previously to work with Neural Networks, the authors have studied other pattern recognition techniques, such as Euclidean and Hamming Distances and Gaussian Mixture Modeling [6],[7],[8]. In order to obtain better results, the authors determined to study the possibility of using Neural Networks [9],[10] for the Recognition Block. The work been done and the results obtained will be detailed in this section.

3.1 Radial Basis Function Neural Networks

The architecture of Radial Basis Networks is based on two layers, where the first one, usually called Radial Basis Layer uses a radial basis function (therefore its name), and the second one uses a pure linear function, being called Linear Layer. The transfer function for a Radial Basis Neuron is defined as:

$$radbas(n) = e^{-n^2} \quad (1)$$

The most important feature of a this kind of networks is that, although may require more neurons than standard feed-forward backpropagation networks (such as Multilayer Perceptron - MLP), they can be designed and trained in a fraction of the time it takes to train standard feed-forward networks.

The architecture of a RBF Neural Network with R components in the input vector, S1 neurons in layer 1 and S2 neurons in layer 2, can be seen in Fig. 4.

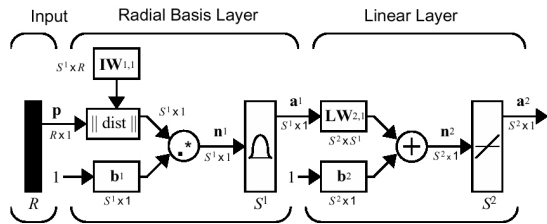


Fig. 4. RBF Network Architecture (taken from [11]).

To use this network in the hand-based biometric recognition system detailed above, it has to be taken into account the following facts:

- The number of components of the feature vector is 25, so $R=25$.
- The number of neurons in the Linear Layer should be equal to the number of users enrolled in the system. In our case, $S2 = 20$.
- The values of $S1$ (number of neurons in the Radial Basis Layer), b (bias input) and IW (weight vector) are determined through the standard zero-error training algorithm used for this kind of networks.

3.2 Experiments and Results

For performing the experiments detailed below, a hand-image database was created, containing at least 10 photographs of each of the 20 users that took part. This database was divided into four groups:

- 1 A number of N samples of each of the 17 first users, for performing the zero-error training. The value of N is one of the parameters studied, and will vary from 3 to 5 samples.
- 2 The rest of the samples of each of those 17 first users, used for simulation.
- 3 N samples of the 3 remaining users. These users are considered as new users entering the system, and this group of samples will be used to train the network.
- 4 The rest of the samples of those 3 users. These samples, together with the ones in group 2, will be used again for simulation.

Main results can be seen in the following table, where the success rates are showed as a function of the number of training vectors used, for each of the two simulation cases: with only the first 17 users, and after including the rest of the users participating in the experiments.

No. training vectors	3	4	5
First 17 users	94.5%	97.3%	98.1%
All users	91.5%	94.5%	96.4%

These results show that with only 5 samples (enough for not making the user feel uncomfortable in the enrolment process), a success above 98% is achieved. Even with 4 training vectors, results are good enough for recognition. Although this results have been satisfactory, the authors consider that further work should be made to increase the success rates, and to analyse the architecture with a larger database, work that is planned to be done in the future.

4 Conclusions

The different steps to achieve a high performance hand-based biometric recognition system have been reported. In our present case, Neural Networks have been used for the recognition process. Better results than the ones reported in [6], [7] and [8] have been obtained. Using Radial Basis Functions, the results are satisfactory, above 98%, but further work should be applied to achieve better figures and to test the system with a larger database.

References

- [1] Jain A.K., Bolle R., Pankanti S., et al. *Biometrics: Personal Identification in Networked Society*. Kluwer Academic Publishers. 1999.
- [2] Jain L.C., Halici U., Hayashi I., Lee S.B., Tsutsui S., et al. *Intelligent Biometric Techniques in Fingerprint and Face Recognition*. CRC Press LLC. 1999.
- [3] Schalkoff R.J. *Digital Image Processing and Computer Vision*. John Wiley & Sons. 1989.
- [4] Jain A.K. *Fundamentals of Digital Image Processing*. Prentice Hall. 1989.
- [5] J hne B. *Practical Handbook on Image Processing for Scientific Applications*. CRC Press LLC. 1997.
- [6] Sanchez-Reillo R. and Gonzalez-Marcos A. "Access Control System with Hand Geometry Verification and Smart Cards". *Proceedings of the 33rd Annual 1999 International Carnahan Conference on Security Technology*. Madrid, Spain, Oct, 1999. pages 485-487.
- [7] Sanchez-Reillo R., Sanchez-Avila C., and Gonzalez-Marcos A. "Multiresolution Analysis and Geometric Measure for Biometric Identification". *CQRE Secure Networking*. D sseldorf, Germany, 1999.
- [8] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos. *Biometric Identification through Hand Geometry Measurements*. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, n° 10, Oct. 2000. pp. 1168-1171.
- [9] Haykin S. *Neural Networks: A Comprehensive Foundation*. Prentice Hall. 1994.
- [10] Sch r mann J. *Pattern Classification: A Unified View of Statistical and Neural Approaches*. John Wiley & Sons, Inc. 1996.
- [11] Demuth H. and Beale M. *Neural Network Toolbox: For Use with Matlab®*. Mathworks. 1998.

Hand Recognition Using Implicit Polynomials and Geometric Features

Cenker Öden, Aytül Erçil, Vedat Taylan Yıldız,
Hikmet Kırmızıtaş, and Burak Büke

Boğaziçi University, Alper Atalay BUPAM Laboratory
80815, Istanbul, Turkey
{oden,ercil,yildizve,kirmizit,bukeb}@boun.edu.tr

Abstract. Person identification and verification using biometric methods is getting more and more important in today's information society; resulting in increased utilization of systems that combine high security and ease of use. Hand recognition is a promising biometric that is being used in low-level security applications for several years. In this work, implicit polynomials, which have proven to be very successful in object modeling and recognition, have been proposed for recognizing hand shapes and the results are compared with existing methods.

1 Introduction

As the personal and institutional security requirements increase, a person has to remember lots of passwords, pin numbers, account numbers, voice mail access numbers and other security codes. In the future, biometric systems will take the place of this concept since it is more convenient and reliable. Trying to come up with a all-purpose, or at least multi-purpose, personal identifier is what the art and science of biometrics is all about. A broad variety of physical characteristics are now being tested to determine the potential accuracy and ultimate consumer acceptance of their biometric measurement as personal identification standards. Up to now, biometric properties like fingerprint, face, voice, handwriting and iris were the subjects of many research efforts and used in different types of identification & verification systems. But the main reason of increased interest in this research area is that as the technology develops, these kinds of systems are more likely to run on the personal devices such as mobile phones and laptops [3].

In many access control systems like border control, personnel follow-up, important point is verification rather than identification. To protect the individual, verification systems are more suitable than highly distinctive biometric systems (iris, fingerprint) [3,7]. Hand recognition systems are very appropriate for these purposes, because they do not cause anxiety like fingerprint and iris systems. People's ease of acceptance due to their convenience, and their easy and cheap setup are the major superiorities of hand geometry based recognition systems to other systems.

There are many hand recognition systems available, and new algorithms were proposed recently giving better performance. Most of these methods rely mainly on geometric features and use the same logic for feature extraction. In this paper, we propose a new method for recognizing hand shape using implicit polynomials. The performance of the proposed system will be compared with existing methods and a way to combining geometric features with the invariants calculated from implicit polynomial fits of the hand will be studied.

2 Implicit Polynomials

Implicit polynomial 2D curves and 3D surfaces are potentially among the most useful object and data representations for use in computer vision and image analysis because of their interpolation property, Euclidean and affine invariants, as well as their ability to represent complicated objects. There have been great improvements concerning implicit polynomials with its increased use during the late 80's and early 90's [2, 5, 10]. Recently, new robust and consistent fitting methods like 3L fitting, gradient-one fitting, Fourier fitting have been introduced [6,11], making them feasible for real-time applications for object recognition tasks.

The implicit polynomial model is given as:

$$f_n(x, y) = \sum_{0 \leq i, j; i+j \leq n} a_{ij} x^i y^j = a_{00} + a_{10}x + a_{01}y + a_{20}x^2 + a_{11}xy + a_{02}y^2 + \dots + \quad (1)$$

An implicit polynomial can be completely characterized by its coefficient vector:

$$[a_{n0}, a_{n-1,1}, a_{n-2,2}, \dots, a_{0n}, a_{n-1,0}, a_{n-2,1}, \dots, a_{0,n-1}, a_{n-2,0}, \dots, a_{0,n-2}, \dots, a_{10}, a_{01}, a_{00}] \Leftrightarrow f_n(x, y) \quad (2)$$

An implicit polynomial curve is said to represent an object

$$\Gamma_0 = \{(x_i, y_i) | i = 1, \dots, K\} \quad (3)$$

if every point of the shape Γ_0 is in the zero set of the implicit polynomial

$$Z(f) = \{(x, y) | f(x, y) = 0\} \quad (4)$$

The zero set of an implicit polynomial fitted to the data will usually be close to the points of Γ_0 but cannot contain all of them.

The most important and the fundamental problem in implicit polynomials is the fitting problem, namely to find the implicit polynomial function $f(x, y)$, or the corresponding coefficient vector that best represents the object. The fitting procedure has to be robust, which means a small amount of change in the data should not cause a relatively huge amount of change in the coefficients of the fitted implicit polynomial. Traditional fitting methods such as least squares fitting lacked this property, and the slightest change in data set caused dramatic differences in resulting coefficients. New algorithms such as 3L fitting [6], gradient-one fitting [9] and Fourier fitting [9] have this desirable character, enabling us to use implicit polynomials for object recognition tasks in a reliable manner.

The main advantage of implicit polynomials for recognition is the existence of algebraic invariants, which are functions of the polynomial coefficients that do not change after a coordinate transformation. The algebraic invariants that are found by Civi [2] and Keren [4] are global invariants and are expressed as simple explicit functions of the coefficients. Their performance have been tested with different complicated objects and they were found to be very successful in object recognition tasks even in the presence of noise, and missing data [12].

3 Methodology

Despite the fact that commercial systems for hand recognition exist in the market, there aren't many and detailed studies on this field in the literature. However due to the reasons explained above, new methods have been proposed recently [1,3,8]. All of the methods proposed use various geometric features of hand (width height and length of the fingers, hand size, height profile, etc.). In our study, we tried to improve the success of the former methods by using implicit polynomials to model the fingers.

Initially preliminary work was performed on the sample database that has been downloaded from [1], (including eight images from nine persons) and tried our algorithm on these images, which gave a 98% of success in identification, encouraging us for future work. We then formed our own hand database by taking 30 images from 28 people. We used backlighting in order to take robust images independent of lighting conditions. We did not use fixation pegs as the methods we referenced employ; and did not constrain users; the only requirements were to place hands in the backlit area and not to combine fingers. Our main motivation in doing so was the high performance of algebraic invariants independent of scaling and rotation. For preprocessing, a LoG edge detector was applied to the acquired images and then images were enhanced to obtain a single-pixel-width boundary of the hand. 20 of these 30 images were used for training and the rest for test purposes. The image acquired from the prototype system and its processed output is seen Fig. 1 and Fig. 2.



Fig. 1. Original image taken with the setup.

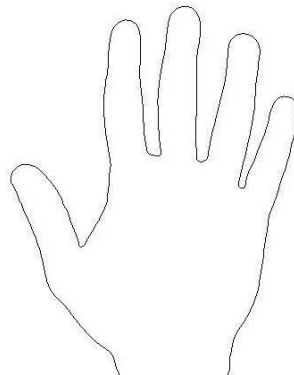


Fig. 2. Processed boundary image.

Geometric features are calculated as seen in Fig. 3 [1,3]. Using the boundary data, a signature analysis was performed to find appropriate points and reliably extract the fingers. A fourth degree implicit polynomial was fitted to each of the fingers both using gradient-one fitting and 3L fitting; resulting coefficient vectors were used to calculate Keren's and Civi's invariants. As previously stated, one of the most useful properties of implicit polynomials is their interpolation property for locally missing data; and we employed this fact during our application as the extracted boundary data for fingers were not connected. Fig. 4 shows how implicit polynomials handle this situation. We observed that Keren's invariants gave better results for all cases, and also gradient-one fitting slightly over performed 3L fitting. Experiment results are summarized in Table 1. Numbers in the parenthesis show the number of features used. For classification purposes, Mahalanobis distance was used.

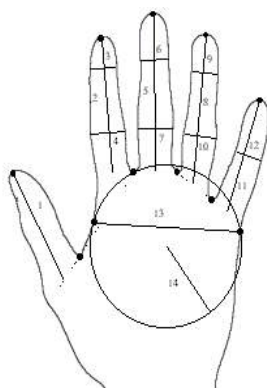


Fig. 3. Geometric features used to construct the feature vector illustrated.

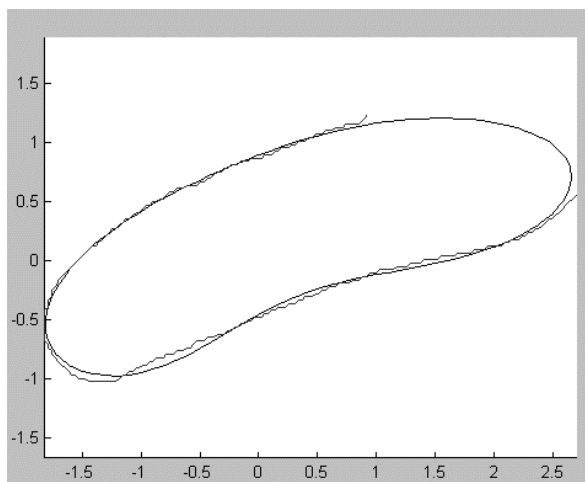


Fig. 4. A 4th degree fit to one of the fingers extracted from boundary data (*closed curve shows the fitted polynomial*).

The first column in the table shows the results of using 16 features. However the second column uses 12 features since it is observed that some features are misleading (e.g. some features for thumb since its geometry is not stable), as a consequence of the image taking policy where minimum constraints were imposed on the user. Third column gives the results obtained using implicit polynomials in all five fingers. Fourth column gives the results by using implicit polynomials in four fingers excluding the thumb. Since 4th degree implicit polynomial fit does not give an accurate fit for the thumbs, identification performance of the invariants is not as good as the geometric features. An alternative approach may be to use 6th degree fits which is more accurate and geometric invariants. This method is expected to increase the success of implicits.

The last column in Table 1 shows the results obtained by combining the features of both methods. As it is clearly seen, while the fusion of the methods increased the identification success above to 95%, the verification rate increased above to 99% and the false acceptance rate decreased down to 1%.

Table 1. Table shows the results obtained using different methods.

	<i>Geometric (16)</i>	<i>Geometric (12)</i>	<i>IP(15)</i>	<i>IP (12)</i>	<i>Geom. + IP (16)</i>
<i>Identification</i>	80%	88%	73%	85%	95%
<i>Verification</i>	89%	98%	84%	90%	99%

4 Conclusions

In this paper, implicit polynomials, have been proposed for recognizing hand shapes and the results are compared with existing methods while researching the best features for identification-verification tasks. The results show that the fusion of invariants from the implicit polynomials and geometric features improve the performance of identification and verification. More accurate 6th degree polynomial fits and geometric invariants have not been tested, but as previous works [12] show, they will most probably give better and more stable results, favoring use of implicit polynomials for hand recognition. A more successful hand recognition system will contribute to the other biometric methods' effectiveness and can widely be used in applications that require low-medium security.

References

1. "HaSiS -A Hand Shape Identification System" www.csr.unibo.it/research/biolab/hand.htm
2. Civi, H., "Implicit Algebraic Curves and Surfaces for Shape Modelling and Recognition," Ph.D. Dissertation, Bogazici University, October 1997.
3. Jain, A.K., Ross, A., and Pankanti, S., "A Prototype Hand Geometry-Based Verification System", in 2nd International Conference on Audio- and Video-based Biometric Person Authentication, Washington D.C., 1999.
4. Keren, D., "Using Symbolic Computation to Find Algebraic Invariants," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 11, pp. 1143-1149, November 1994.
5. Keren, D., Cooper D., and Subrahmonia, J. "Describing complicated objects by implicit polynomials," *IEEE PAMI*, Vol. 16, No. 1, pp. 38-53, January 1994.
6. Lei Z., Blane M.M. and Cooper D.B. "3L Fitting of Higher Degree Implicit Polynomials" In Proceedings of third IEEE Workshop on applications of Computer Vision, Saratosa, FL, December 1996.
7. Miller, B., "Vital Signs of Identity", IEEE Spectrum, February 1994.
8. Sanchez-Reillo, R., Sanchez-Avila and C., and Gonzales-Marcos A., "Biometric Identification Through Hand Geometry Measurements", IEEE Transactions on Pattern Analysis and Machine Intelligence, October 2000.
9. Taşdizen, T., Tarel, J.P., and Cooper, D.B., "Improving the Stability of Algebraic Curves for Applications", IEEE Transactions on Pattern Analysis and Machine Intelligence, March 2000.
10. Taubin, G., "Estimation of planar curves, surfaces and nonplanar space curves defined by implicit equations, with applications to edge and range image segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, No. 11, pp. 1115-1138, November 1991.
11. Ünsalan, C., Erçil, A. "A New Robust and Fast Technique for Implicit Polynomial Fitting", Proceedings of M²VIP, , p. 15-20, September 1999.
12. Yalınz, M., "Algebraic Invariants for Implicit Polynomials Case Study: Patterns in Turkish Hand-woven Carpets", Boğaziçi University M.S. Thesis, 1999.

Including Biometric Authentication in a Smart Card Operating System

Raul Sanchez-Reillo

Carlos III University of Madrid, Dpt. Electric, Electronic and Automatic Engineering
c/Butarque, 15, E-28911 Leganes, Madrid, Spain
rsreillo@ing.uc3m.es

Abstract. One of the most secure medias in where to store personal a sensible data, are smart cards. They can provide security through cryptography and secret keys. Unfortunately, they suffer a lack of security when the card holder identity is to be authenticated. The only way, nowadays, to achieve such a task is through Card Holder Verification Keys, which are closely related to the Personal Identification Number (PIN). The author, aware of this problem, has worked in enabling a Biometric Authentication inside a smart card, in order to provide the same level of security that PINs have, without being able to be copied. After studying several biometric techniques, he has developed, up to today, three of them (based on Voice, Hand Geometry and Iris). Therefore, he has obtained the conclusions needed to integrate the Biometric Authentication into the Operating System of a Smart Card. Using JavaCards, prototypes have been developed, taking several tests to prove the viability of them. Results obtained, specially the ones using the RISC-based JavaCard, show the possibility of launching a commercial product, without going to an expensive masking level of development.

1 Introduction

In many Information Technology (IT) applications, data should be stored in a distributed way, due to the fact that some sensible and/or personal information of the final user should be kept with him. One of the possibilities is to give each user an identification token, which stores his personal information, as well as any other information, requested by the system designed. Some of that information, such as financial data, health care record, etc., should be protected by the user who owns de card (i.e. the cardholder). This protection is usually made with CHV-Keys (i.e., Card Holder Verification Keys), which are closely related to the user's PIN. In a smart card, security with this kind of keys can involve allowing or denying not only access to some information, but also the possibility of performing some operations, such as a debit from an electronic purse. But this protection is improved as a smart card have the possibility to block that access for life if a determined number of wrong presentations of the PIN, or other secret keys, has been reached.

Unfortunately PINs, as passwords, can be copied by inspecting the cardholder movements as he enters his number in, for example, an ATM. The only way to perform a real user authentication is through biometrics [3]. But biometric verification

cannot be found in any commercial smart card nowadays. The only efforts being applied in this line is to store the user's template inside a smart card, protected with Administrative Keys, and extracted from the card by the terminal to perform there the verification.

This paper presents the works being carried by the author in order to achieve a new authentication system, following his works in [10]. In this authentication system, the user's template is stored inside a smart card, with the same level of protection as any other key (i.e. not allowing the reading of it, having the possibility to block temporally or for life, etc.). Then, when the user wants to authenticate himself to the card, he asks for such an operation, giving one biometric sample, which is verified inside the card. Whether the verification is positive, the card allows the access to the biometrically protected information and/or operations.

To develop the prototype, the author has studied different biometric techniques, and developed three of them, which will be mentioned in the next paragraph. Then the results obtained with those techniques are compared in order to obtain restrictions to its integration inside a smart card. After that, the way the integration has been performed is explained, giving the results obtained with three different Open Operating System smart cards. At the end of the paper overall conclusions will be given.

2 Different Biometric Techniques Developed

In order to obtain general conclusions about the possibility of integrating biometric authentication inside a smart card, the author has deeply studied several techniques. From these, the author has also developed three of them: Speaker Recognition, Hand Geometry and Iris Identification. This section is intended to give an overall introduction to each of these three techniques.

It is important to note that the results that are going to be given about these three techniques, specially those referring the time spent in the algorithms, have been obtained with the execution of the algorithms in a PC through the programs developed in a mathematical development software called MATLAB. Unfortunately, this development system, does not obtain a good performance in computation time, and those results can be easily improved by coding those algorithms in C or C++.

2.1 Speaker Recognition

In this technique, the user's voice is captured with a microphone, then pre-processed and passed through a block that is in charge of extracting the cepstrum coefficients. From all the verification methods being applied to the human voice for Speaker Recognition [1], the author has used Gaussian Mixture Models (GMM), due to its lower memory requirements and better results [4].

In order to obtain the user's template, 60 seconds of continuous speech from the user should be taken to train the GMM. Once trained, only 3 seconds of the user's speech are needed to perform the biometric verification.

The user's template is 175 bytes long, while the sampling utterance that is passed to the verification algorithm, i.e. the GMM, is 3600 bytes long. Considering the computation time spent, the enrollment phase (i.e. when the user's template is extracted) last for about 1100 seconds, while the time needed for the verification is 16 seconds. Unfortunately, the error rates obtained for a 1-time verification process (i.e. if the first verification fails, the user is rejected) are much above 10%, obtaining an Equal Error Rate (EER) of 20.3%.

2.2 Hand Geometry

As seen in Figure 1, this technique is based on taking a photograph of the user's hand [3][5][7][8]. The hand is placed on a platform painted in blue -to increase contrast with all kind of skins-. Six tops are placed on the platform to guide the hand in front of the medium resolution digital camera. A photograph is taken and the hand contour is extracted. A deeper description of the system can be found in [11].

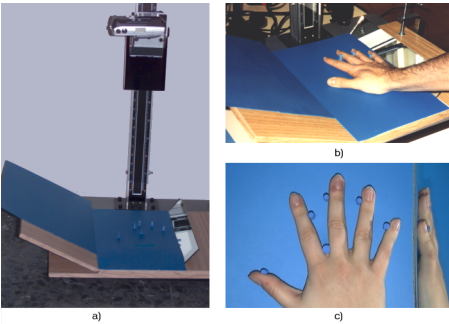


Fig. 1. Hand Geometry prototype developed. a) General view of the system; b) placement of the hand; c) photograph taken.

Several measurements are performed on the contour, obtaining a feature vector of 25 bytes. With 5 photographs from his hand, the user's template is computed, depending on the verification method used. The methods tested have been: a) the Euclidean Distance; b) the Hamming Distance; and c) GMM. The results obtained can be seen in the following table:

	Template size	Sample size	Enrollment time	Verification time	EER (in %)
Euclidean D.	25 b	25 b	37 s	7.5 ms	16.4
Hamming D.	50 b		37.1 s	10 ms	8.3
GMM	510 b		37.5 s	30 ms	6.5

2.3 Human Iris

From all the biometric techniques known today, the most promising is iris identification, due to its low error rates, nearly null False Acceptance Rate (FAR) and without being invasive [2],[6],[7]. The system developed, based on Daugman's work [2], takes a high-resolution photograph of the user's eye, and pre-processes it to extract the inner and outer boundaries of the iris. After that, Gabor filters are used to extract the feature vector, which is 233 bytes long. Deeper information about the prototype developed by the author, can be found in [9].

The user's template is obtained from a single sample, and the Hamming Distance is used to verify the samples with the template stored. Both, the template and the sample size are 233 bytes. The enrollment time is determined by the pre-processing and feature extraction blocks, which can be optimized from the results obtained in the prototype developed (142 seconds). The verification last 9 ms. The False Rejection Rate for 1 time verification is 3,51% with a null FAR.

3 Comparison Among the Results Obtained

From the results achieved with the above-mentioned techniques, several conclusions can be obtained focusing on the possibility of integrating biometric authentication into a smart card:

- Due to the sample vector size and verification times, Speaker Recognition is not considered as a viable technique for the purposes stated in this paper. Also, the error rates achieved should be improved.
- Error rates obtained with the Euclidean Distance in the Hand Geometry technique are not good enough to consider it for a medium/high security system. However, due to its simple verification algorithm, it could be interesting to integrate this technique in a 3-time retry authentication method, improving, therefore, the error rates.
- Hand Geometry and Iris Identification, both with Hamming Distances, seem to fit perfectly with the purposes of this paper.
- Hand Geometry with GMMs achieves really good results. However, its computation cost and the need of using floating-point operations, will disable -as stated below- the possibility of integrating it inside a smart card. Further work should be applied to enable this possibility.

4 Placing the Biometric Authentication inside a Smart Card

In order to perform the integration, two possibilities exist: a) building a whole new mask, which is a very high cost processing but enable to achieve the best results; b) using an Open Operating System smart card, such as a Java Card [12], which is a low cost process but has certain constraints given by the platform used.

The author has chosen this second way to implement the biometric authentication inside a smart card, using three different products from two different manufacturers. There is a main difference between one of the cards of one manufacturer and the other two cards used, and is the processor architecture used. While the other two, use a 16-bit CISC processor, that one uses a 32-bit RISC processor.

The prototypes implemented have been built with the following elements:

- Some environmental variables that indicate *blocking*, *successful verification*, *number of maximum and remaining tries*, *possibility of unblocking*, etc. of the biometric template.
- `create_template_file` function, which creates the file where the user's template is going to be stored, giving some parameters, such as the algorithm that applies to perform the verification.
- `write_template_file` function, that writes the data corresponding to the user's template, including the maximum number of erroneous verifications allowed. This function is based on the `UPDATE_BINARY` standard function.
- `personal_authentication` function. This function has being coded as the standard function accepted by the industry.
- `template_unblock` function, which can be executed only if the template file is defined as having the possibility of being unblocked. This will depend on the level of security needed by the application where the card is going to be used.
- Several testing commands (APDUs) to verify the right functioning of the prototype.

All these elements have been developed following all the available standards, such as the ISO 7816. The results obtained with the three card mentioned above -sorted in two categories according to the architecture of its processor- can be seen in the following table:

	RISC	CISC	Units
Size of the prototype code	2121	1511	b
Authentication Time (Hand Geometry, Euclidean D.)	5.11	121	ms
Authentication Time (Hand Geometry, Hamming D.)	30.1	127	ms
Authentication Time (Iris Identification)	7.11	1230	ms

Three main considerations should be made from these results. The first one is that Hand Geometry with GMMs has not been covered. This has been impossible because JavaCard specifications do not accept floating-point operations. The second consideration is that times obtained with the RISC processor are much lower than the ones obtained with the CISC processor, enabling much sophisticated verification methods in a near future. Unfortunately the cost of the RISC processor manufacturing has led to forget for a while this version of the card. The last consideration is that optimizing the binary functions in the JavaCard platform can easily lower verification time obtained with the CISC processor, which is higher than an acceptable time of half a second.

5 Conclusions

The integration of Biometric Authentication inside a Smart Card Operating System has been presented. Three biometric techniques have been studied to analyze their viability. These techniques have been Speaker Recognition, Hand Geometry and Iris Identification. The results obtained show the viability of the prototypes, using Biometrics as a Card Holder Verification method, therefore, improving user authentication in smart card based applications. Better results can be obtained building a new smart card mask, instead of using an Open Operating System card, such as the JavaCards used in the prototypes developed. If this last possibility is not possible, results with the RISC based JavaCard are good enough for a commercial product. Further efforts will be applied to integrate another biometric techniques in the prototypes developed, such as fingerprint or facial recognition.

References

- [1] J.P. Campbell, Jr. "Speaker Recognition: A Tutorial". Proceedings of the IEEE, vol. 85, n° 9, pp. 1437-1462, Sep. 1997.
- [2] J.G. Daugman. "High Confidence Visual Recognition of Persons by a Test of Statistical Independence". IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 15, n° 11, Nov. 1993. pp. 1148-1161.
- [3] A.K. Jain, R. Bolle, S. Pankanti, et al. Biometrics: Personal Identification in Networked Society. Kluwer Academic Publishers. 1999.
- [4] D.A. Reynolds, R.C. Rose. "Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models". IEEE Trans. on Speech and Audio Processing, vol. 3, num. 1, pp. 72-83, Jan. 1995.
- [5] R. Sanchez-Reillo and A. Gonzalez-Marcos. "Access Control System with Hand Geometry Verification and Smart Cards". Proc. 33rd Annual 1999 International Carnahan Conference on Security Technology. Madrid (Spain), 5-7 Oct, 1999. pp. 485-487.
- [6] R. Sanchez-Reillo, C. Sanchez-Avila, and J.A. Martin-Pereda. "Minimal Template Size for Iris-Recognition". Proc. of the First Joint BMES/EMBS Conference. Atlanta (U.S.A.), 13-16 Octubre, 1999. p. 972.
- [7] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos. "Multiresolution Analysis and Geometric Measure for Biometric Identification". Secure Networking - CQRE [Secure]99. Nov/Dec, 1999. Lecture Notes in Computer Science 1740, pp. 251-258.
- [8] R. Sanchez-Reillo and A. Gonzalez-Marcos. "Access Control System with Hand Geometry Verification and Smart Cards". IEEE Aerospace and Electronic Systems Magazine, vol. 15, n° 2, Feb. 2000. pp. 45-48.
- [9] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos. "Improving Access Control Security using Iris Identification". Proc. 34rd Annual 2000 International Carnahan Conference on Security Technology. Ottawa (Canada), 23-25 Oct, 2000. pp. 56-59.
- [10] R. Sanchez-Reillo. "Securing Information and Operations in a Smart Card through Biometrics". Proc. 34rd Annual 2000 International Carnahan Conference on Security Technology. Ottawa (Canada), 23-25 Oct, 2000. pp. 52-55.
- [11] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos. Biometric Identification through Hand Geometry Measurements . IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, n° 10, Oct. 2000. pp. 1168-1171.
- [12] Sun Microsystems, Inc. <http://java.sun.com/products/javacard>.

Hybrid Biometric Person Authentication Using Face and Voice Features

Norman Poh and Jerzy Korczak

LSIIT, ULP-CNRS, Bld S. Brant, 67400 Illkirch, France
{poh,jjk}@dpt-info.u-strasbg.fr

Abstract. In this paper, a hybrid person authentication prototype integrating multiple biometric devices is presented. This prototype is based on several levels of abstractions: data representations, vectors and classifiers. Frontal face and text-dependent voice biometrics are chosen to authenticate a user. For each of the biometric feature, an extractor, a classifier and a simple negotiation scheme have been designed. An extractor is made up of a sequence of operators which themselves are made up of signal processing and image processing algorithms. The face information is extracted using moments and the short speech information is extracted using wavelets. The extracted information, called vectors, is classified using two separate multi-layer perceptrons. The results are combined using a simple logical negotiation scheme. The prototype has been tested and evaluated on real-life databases.

1 Introduction

The current authentication systems are characterized by an increasing interest in biometric techniques. Among these techniques are face, facial thermogram, fingerprint, hand geometry, hand vein, iris, retinal pattern, signature and voiceprint. All these methods have different degrees of uniqueness, permanence, measurability, performance, user's acceptability and robustness against circumvention [4].

The latest research on multimodal biometric systems shows that we may improve the incompleteness of any unimodal biometric system. Brunelli et al. have proposed two independent biometric schemes by combining evidence from speaker verification and face recognition [1]. Dieckmann et al. have proposed an abstract level fusion scheme called 2-from-3-approach which integrates face, lip motion and voice based on the principle that a human uses multiple clues to identify a person [2]. Kittler et al. have demonstrated the efficiency of an integration strategy that fuses multiple snapshots of a single biometric property using a Bayesian framework [5]. Maes et al. have proposed to combine biometric data, e.g. voice-print, with non-biometric data, e.g., password [6]. Jain et al. have proposed a multimodal biometric system design which integrates face, fingerprint and speech to make a personal identification [4].

The goal of this project is to design a hybrid biometric system that is independent of any biometric device. An abstraction scheme is proposed that can combine any biometric feature and facilitates the integration of new biometric features. By abstrac-

tion, we group these techniques into 1D, 2D or 3D recognition problems. For example, voice-print and signature acoustic is considered as a 1D pattern recognition problem, mug-shot face, facial thermogram, fingerprint, hand geometry, hand vein, iris, retinal pattern can be considered as a 2D or image recognition problem and face can be considered as a 3D or object recognition problem. Having classified these problems, the objective is to define a set of basic operations that work on 1D, 2D and 3D problems. These operations constitute the building blocks of extractors that can be defined to conceive a set of independent extractors either statistically compiled or dynamically linked. Each extractor produces its own type of feature vector. The produced vector represents the biometric feature that can discriminate one person from another. The vector will be classified by its proper classifier. To combine the different results of classifiers, various negotiation strategies can be used [8].

The second section of this paper discusses the details of biometric authentication techniques, namely the face and the voice extractors, and neural networks as classifiers with a logical negotiation scheme. The third section discusses the databases and experiments protocol and the obtained results.

2 Biometric Authentication Methods

2.1 Face Authentication

In face recognition, problems are caused by different head orientations. So, if only the information around the eyes is extracted, then head orientations will not contribute to the errors. Of course, in doing so, other face information will be lost. Nevertheless, as a start, we opt for this simpler approach [8].

Firstly, a face image is captured using a web camera. A face is then detected using template matching. The user, however, has to move into the template rather than the template moving to search the face location. Eyes are then automatically localized using a combination of histogram analysis, round mask convolution and a peak-searching algorithm.

Moments are used to extract the eye information because it is a simple yet powerful extractor. Normalized central moments are invariant to translation, rotation and scaling. A moment of order $p+q$ of an image f_{xy} of N by N pixels with respect to a center (\bar{x}, \bar{y}) is given in Eq. 1 (more details can be obtained from [3].)

$$M_{pq} = \sum_x^{N-1} \sum_y^{N-1} f_{xy} (x - \bar{x})^p (y - \bar{y})^q \quad (1)$$

Instead of working on the RGB (red green blue) color space, we worked on the HSI (hue saturation intensity) color space as well. For each eye, a pair of moments is extracted from the green, blue, hue, saturation and intensity color space. These parameters make a vector of 10 dimensions for each eye. The magnitude of each item in the eye vector is compressed using the logarithmic function and then normalized into the range zero and one. Fig. 1 illustrates the idea.

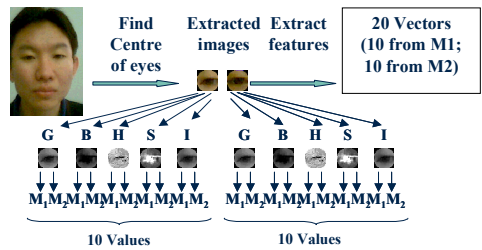


Fig. 1. Eye feature extraction using moments.

2.2 Text Dependent Speaker Authentication

The front end of the speech module aims to extract the user dependent information. It includes three important steps: speech acquisition, detection and extraction. In general, the user s vocal password is sampled via a microphone at 8 kHz over a period of 3 seconds. In the second step, the presence of speech is then detected and then extracted using the Morlet wavelet [7].

By convoluting the wavelets with a speech signal, several scales of wavelet coefficients are obtained. The magnitude of wavelet coefficients is proportionate to the variation of signals. High magnitude of wavelet coefficients at a scale means a high variation change. Based on this information, it is possible to segment the speech signal and then used the segmented wavelet coefficients as a vector feature.

In our experiments, a wavelet transform on a speech signal of 3 seconds gives 8 analyzable scales. By using signal-to-noise analysis on the wavelet coefficients scale, we were able to determine that wavelets of scale-1, 2, 3 and 4 are more significant than other scales. Each of these scales is then truncated, normalized and then sampled before being merged to form a vector of 64 values (see Fig. 2). Through this sampling process, some important data could be lost. Such data reduction is necessary to make sure that the final vector is small enough to train the neural network.

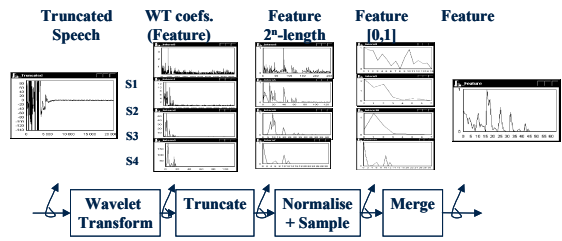


Fig. 2. Voice feature extraction using wavelet transform.

2.3 Classifier and Negotiation

A Multi-Layer Perceptron (MLP) is used for each type of face and voice vectors because it is robust against noise and efficient. In this project MLP is considered as a universal classifier. An authorized person has his proper face and voice MLPs. Each of the MLPs is trained using the classical back-propagation algorithm. The output of each of the MLPs is true if the output neuron activation is more than an optimized threshold.

Each of the MLPs is trained several times and only the best observed MLP is retained. The auto-selection scheme, in general, is performed by evaluating the minimum cost error committed by the MLP evaluated on a validation set [8] (see experiment protocol below). As for the fusion of decision of two classifiers, instead of using complicated fusion scheme, we opt for a logical AND operation.

3 Test Results

3.1 The Database

Two databases have been created for validation of our approach. The first database contains 30 persons. Each person has 10 face images and 10 sessions of speech recordings. The second database has 4 persons and each person has 100 face images and 100 sessions of speech recordings. The same amount of vectors is extracted using the raw biometric data. For the first and second databases, there are, therefore, $30 \text{ persons} \times 10 \text{ sessions} \times 2 \text{ biometric types} = 600 \text{ vectors}$ and $4 \times 100 \times 2 = 800 \text{ vectors}$ respectively. The first database is aimed at simulating the real situation whereby an access point provides biometric-enabled check for 30 people. The second database is created so that more data is made available to train the MLP.

The database was acquired using a Creative WebCam and a standard PC microphone. The front view face image captured has a dimension of 150×225 pixel in RGB color and the voice is sampled at 8 kHz over 3 seconds. The biometric data was captured within one visit of the person to minimize the cost needed to capture large amount of data.

3.2 The Experiments Protocol

The database of features is divided into training set, validation set and test set. Each of them created according to the cross-validation protocol. The training set is the data used directly to train the MLPs, i.e., changing the weight connections, while the validation set is used to calibrate the threshold and control the training, i.e., to determine the stopping condition and select the best trained MLP from a population of MLPs, and the test set is used exclusively to test the trained MLP. The training:validation:test ratio are 3:1:1 and 5:2:3 for the first and second databases respectively.

Two error measures are used: *False Acceptance Rate* (FAR) and the *False Rejection Rate* (FRR). FAR and FRR are functions of a threshold that can control the trade-off between the two error rates [8].

The performance of the authentication system can be measured by plotting a Receiver Operating Characteristics curve (ROC), which is a plot of FRR versus FAR. The point on the ROC defined by FAR=FRR is the Equal Error Rate point (EER). The crossover accuracy is measured as $1/EER$, which can be interpreted as how many user the system can distinguish correctly before an error is committed. It should be noted that FAR, FRR and EER are data-dependent measurement and often does not reflect the real statistics.

3.3 Experiment Results

From the first database, five samplings of ROC are examined and their median is then plotted in Fig. 3(a). It can be observed that the voice MLP performs better than the face MLP because the ROC of the voice MLP lays nearer to the origin. However, their EERs are about the same, i.e., 0.10. By analyzing the density of FAR, out of 30 of the combined MLPs for 30 persons, 66.7% of them achieved FRR=0, 16.0% of them achieved FRR=0.25 (1 false rejection out of the combined 2 face vectors \times 2 voice vectors) and 17.3% of them achieved FRR=0.50. As for the FAR, 98.7% of them achieved FAR=0 and 1.3% of them achieved FAR=0.009.

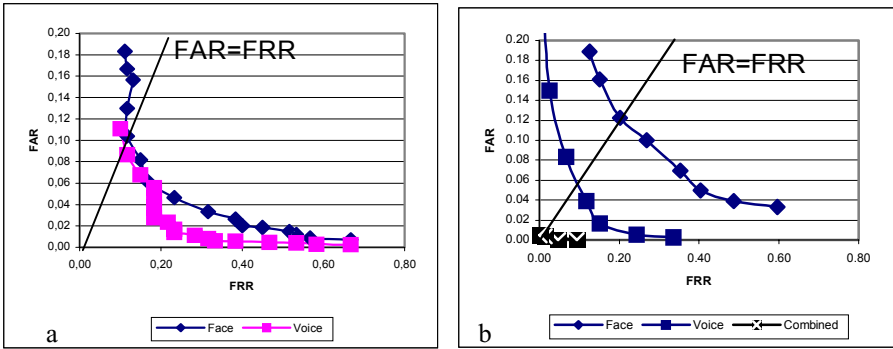


Fig. 3 (a) Median of 5 ROC based on 30 persons (database I) **(b)** Median of 5 ROC based on 4 persons (database II).

In Fig. 3(b), the EER for the overall face MLP is about 0.15 while the EER for the overall voice MLP is around 0.07. The weak recognition rate of the voice MLP may be caused by the significant loss of information during the sampling of wavelet coefficients. Fig. 3(b) shows that there is a significant gain of performance when the two features are combined even though the EER is not measurable.

4 Conclusion

The prototype of hybrid person authentication system using a vector abstraction scheme and learning-based classifiers is a promising technique. From both the design and research points of view it is a valuable tool because it greatly facilitates the search of new extractors and classifiers. The extractors are made up of a sequence of operators which themselves are made up of signal processing and image processing algorithms. From a biometric data, an extractor extracts discriminative information and represents it in the form of a vector. A vector is then classified using its proper classifier that is made up of a set of learning-based matching algorithms.

The frontal face and text-dependent voice biometrics are chosen in this project. The classifiers for each of the biometric data are Multi-Layer Perceptrons combined by a logical decision-merging scheme. The experiments have confirmed that a multi-modal approach is better than any single modalities. The developed prototype depends on the performance of the extractors, i.e., how discriminative they are in extracting user-dependent information and the state of classifiers, i.e., how they are adequately trained and configured (the threshold value) before putting to use.

Our future directions will be to test the quality of vectors using vector quantization or self-adapting networks that can measure inter-class and intra-class distance in order to search for the best discriminative extractor for a given application. It should be underlined that the learning-based classifiers need a large amount of biometric data not only to train the system but also to test the system independently.

References

1. Brunelli, R. and Falavigna, D.: "Personal identification using multiple cues," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 10, pp. 955-966, 1995.
2. Dieckmann, U., Plankensteiner, P., and Wagner, T.: SESAM: A biometric person identification system using sensor fusion, In *Pattern Recognition Letters*, Vol. 18, No. 9, pp. 827-833, 1997.
3. Gonzalez, R., and Woods, R.: "Digital Image Processing", 2nd edition, Addison-Wesley, 1993.
4. Jain, A., Bolle, R., and Pankanti, S.: BIOMETRICS: Personal identification in networked society, 2nd Printing, Kluwer Academic Publishers, 1999.
5. Kittler, J., Li, Y., Matas, J., and Sanchez, M.U.: Combining evidence in multi-modal personal identity recognition systems, In *Proc. 1st Int. Conf. On Audio Video-Based Personal Authentication*, pp. 327-344, Crans-Montana, 1997.
6. Maes S. and Beigi, H.: "Open sesame! Speech, password or key to secure your door?", In *Proc. 3rd Asian Conference on Computer Vision*, pp. 531-541, Hong Kong, 1998.
7. Masters, T.: Signal and image processing with neural networks: A C++ Sourcebook, Academic Press, 1994.
8. Poh, N. and Korczak, J.: Biometric Authentication System, Res. Rep. LSIIT, ULP, 2001.

Information Fusion in Biometrics

Arun Ross¹, Anil K. Jain¹, and Jian-Zhong Qian²

¹ Michigan State University, East Lansing, MI, USA 48824
{rossarun,jain}@cse.msu.edu

² Siemens Corporate Research, Princeton, NJ, USA 08540
qian@scr.siemens.com

Abstract. User verification systems that use a single biometric indicator often have to contend with noisy sensor data, restricted degrees of freedom and unacceptable error rates. Attempting to improve the performance of individual matchers in such situations may not prove to be effective because of these inherent problems. Multimodal biometric systems seek to alleviate some of these drawbacks by providing multiple evidences of the same identity. These systems also help achieve an increase in performance that may not be possible by using a single biometric indicator. This paper addresses the problem of *information fusion* in verification systems. Experimental results on combining three biometric modalities (face, fingerprint and hand geometry) are also presented.

1 Introduction

The performance of a biometric system is largely affected by the reliability of the sensor used and the degrees of freedom offered by the features extracted from the sensed signal. Further, if the biometric trait being sensed or measured is noisy (a fingerprint with a scar or a voice altered by a cold, for example), the resultant confidence score (or matching score) computed by the matching module may not be reliable. Simply put, the matching score generated by a noisy input has a large variance. This problem can be alleviated by installing multiple sensors that capture different biometric traits. Such systems, known as *multimodal biometric systems* [1], are expected to be more reliable due to the presence of multiple pieces of evidence. These systems are able to meet the stringent performance requirements imposed by various applications. Moreover, it will be extremely difficult for an intruder to violate the integrity of a system requiring multiple biometric indicators. However, an integration scheme is required to fuse the information churned out by the individual modalities. In this work we address the problem of *information fusion* by first building a multimodal biometric system and then devising various schemes to integrate these modalities. The proposed system uses the fingerprint, face, and hand geometry features of an individual for verification purposes.

2 Fusion in Biometrics

Figure 1 illustrates the various levels of fusion that are possible when combining multiple biometric systems: (a) fusion at the feature extraction level, where features extracted using multiple sensors are concatenated, (b) fusion at the confidence level, where matching scores reported by multiple matchers are combined [2], and (c) fusion at the abstract level, where the accept/reject decisions of multiple systems are consolidated [3].

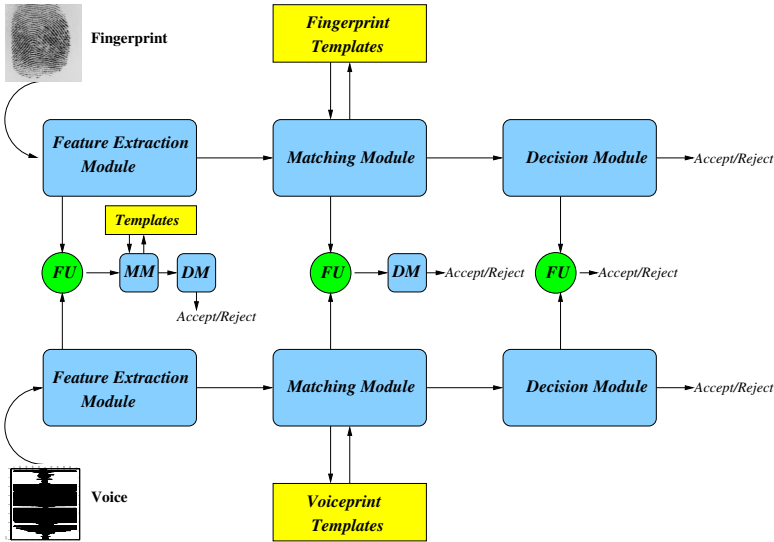


Fig. 1. A bimodal biometric system showing the three levels of fusion; FU: Fusion Module, MM: Matching Module, DM: Decision Module.

Fusion in the context of biometrics can take the following forms: (i) Single biometric multiple classifier fusion, where multiple matchers on a single biometric indicator are combined [1]. (ii) Single biometric multiple matcher fusion, where scores generated by multiple matching strategies are combined [4]. (iii) Multiple biometric fusion, where multiple biometrics are utilized [5], [6], [7].

An important aspect that has to be dealt with is the normalization of the scores obtained from the different domain experts [8]. Normalization typically involves mapping the scores obtained from multiple domains into a common framework before combining them. This could be viewed as a two-step process in which the distributions of scores for each domain is first estimated using robust statistical techniques and these distributions are then scaled or translated into a common domain.

3 Experiments

A brief description of the three biometric indicators used in our multimodal verification system is given below. Our experiments deal with combining information at the representation and confidence levels, and not at the abstract level. There is very little information available at the abstract level, and a simple voting scheme would be expected to do well at this level [4].

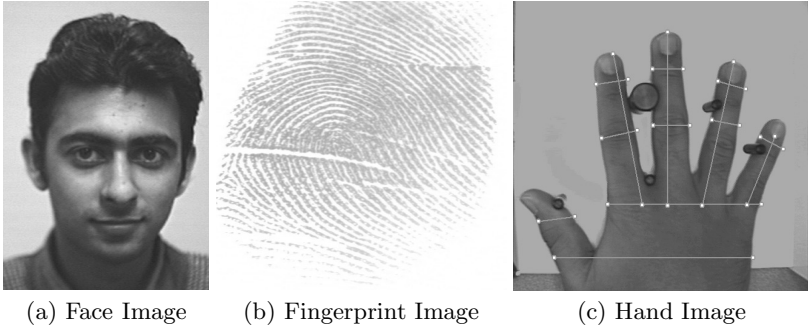


Fig. 2. The three biometric indicators used in our experiments.

1. Face Verification:

Grayscale images of a subject's face were obtained using a Panasonic video camera. The eigenface approach was used to extract features from the face image [4]. In this approach a set of orthonormal vectors (or images) that span a lower dimensional subspace is first computed using the principal component analysis (PCA) technique. The feature vector of a face image is the projection of the (original face) image on the (reduced) eigenspace. Matching involves computing the Euclidean distance between the coefficients of the eigenface in the template and the eigenface for the detected face.

2. Fingerprint Verification:

Fingerprint images were acquired using a Digital Biometrics sensor at a resolution of 500 dpi. The features correspond to the position and orientation of certain critical points, known as minutiae, that are present in every fingerprint. The matching process involves comparing the two-dimensional minutiae patterns extracted from the user's print with those in the template [11].

3. Hand Geometry Verification:

Images of a subject's right hand were captured using a Pulnix TMC-7EX camera. The feature extraction system computes 14 feature values comprising of the lengths of the fingers, widths of the fingers and widths of the palm at various locations of the hand [10]. The Euclidean distance metric was used to compare feature vectors, and generate a matching score.

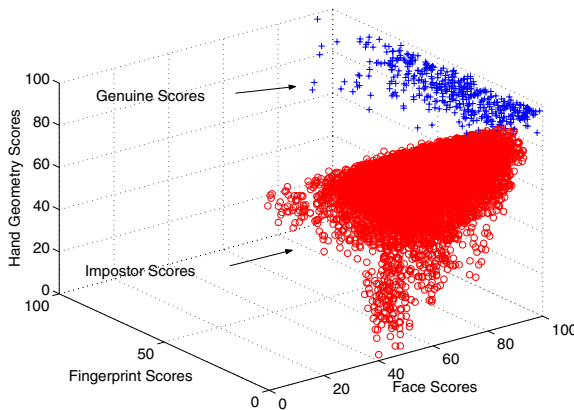


Fig. 3. Scatter Plot showing the genuine and impostor scores. The points correspond to 450 genuine scores (+) and 11·025 impostor scores (o).

The database for our experiments consisted of matching scores obtained from the face, Fingerprint and hand geometry systems. However, data pertaining to all three modalities were not available for a single set of users. The mutual independence of these three biometric indicators allows us to collect the biometric data individually and then augment them. The Fingerprint and face data were obtained from user set I consisting of 50 users. Each user was asked to provide 9 face images and 9 Fingerprint impressions (of the same Fnger). The hand geometry data was collected separately from user set II also consisting of 50 users (some users from set I were present in set II). Each user in set I was randomly paired with a user in set II. 450 genuine scores and 22·050 impostor scores were generated for each of the three modalities. All scores were mapped to the range [0·100]. A score vector - (x_1, x_2, x_3) - represents the scores of multiple matchers, with x_1 , x_2 and x_3 corresponding to the scores obtained from the 3 modalities. The three-dimensional scatter plot of the genuine and impostor scores is shown in Figure 3. The plot indicates that the two distributions are reasonably well separated in 3-dimensional space; therefore, a relatively simple classifier should perform well on this dataset.

1. Sum Rule:

The sum rule method of integration takes the weighted average of the individual score values. This strategy was applied to all possible combinations of the three modalities. Equal weights were assigned to each modality as the bias of each matcher was not available. Figure 4(a) shows the performance of the sum rule using only two modalities, and Figure 4(b) shows the performance using all three modalities.

2. Decision Tree:

The C5.0 program was used to generate a decision tree from the training set of genuine and impostor score vectors. The training set consisted of 11·025 impostor score vectors and 225 genuine score vectors. The test set consisted of the same number of independent impostor and genuine score vectors.

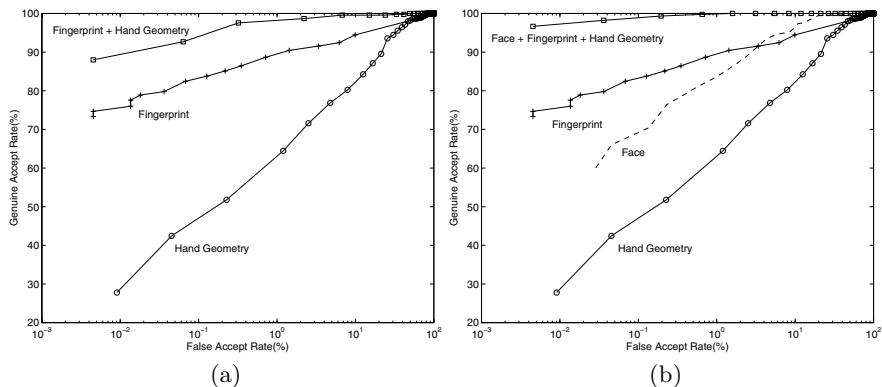


Fig. 4. ROC curves using the sum rule: (a) Combining Fingerprint and hand geometry scores, (b) Combining Fingerprint, face and hand geometry scores.

Table 1(a) shows the performance of the $C5\&0$ decision tree on one such test set.

3. Linear Discriminant Function:

Linear discriminant analysis of the training set helps in transforming the 3-dimensional score vectors into a new subspace that maximizes the between-class separation. The test set vectors are classified by using the minimum Mahalanobis distance rule (with the assumption that the two classes have unequal covariance matrices). Table 1(b) shows the confusion matrix resulting from using this quadratic decision rule on the test set.

Table 1. Confusion matrices showing the performance of the (a) $C5\&0$ Decision Tree, and (b) Linear Discriminant classifier, on an independent test set.

	Genuine	Impostor
Genuine	203	22
Impostor	4	11,021

	Genuine	Impostor
Genuine	225	0
Impostor	72	10,953

(a) $C5.0$ Decision Tree. (b) Linear Discriminant classifier.

The experimental results show that the sum rule performs better than the decision tree and linear discriminant classifiers. The FAR of the tree classifier is 0.036% ($\approx 0.03\%$) and the FRR is 9.63% ($\approx 0.03\%$). The FAR of the linear discriminant classifier is 0.47% ($\approx 0.3\%$) and its FRR is 0.00%. The low FRR value in this case is a consequence of overfitting the genuine class which has fewer samples in both test and training sets. The sum rule that combines all three scores has a corresponding FAR of 0.03% and a FRR of 1.78% suggesting better performance than the other two classifiers. It has to be noted that it is not possible to fix the FRR (and then compute the FAR) in the case of the decision tree and linear discriminant classifiers.

We also investigated the integration of multiple biometric modalities at the representation level. The face and Fingerprint feature vectors were augmented to create a higher dimensional feature vector. A texture-based feature set, as

opposed to a minutiae-based set, was used to represent fingerprints in this case [12]. The normalized feature vector was used to represent the identity of a person. Initial experiments show that this augmented feature vector performs better than combining scores at the confidence level (sum rule). We are conducting more extensive experiments to examine fusion at the representation level.

4 Conclusion

This paper provides initial results obtained on a multimodal biometric system that uses face, fingerprint and hand geometry features for verification. All the three fusion schemes (at the confidence level) considered here provide better verification performance than the individual biometrics. It would be instructive to study other datasets involving a larger number of users with additional biometric indicators. Towards this end, we are in the process of collecting data corresponding to four biometric indicators - fingerprint, face, voice and hand geometry - from a larger number of users.

References

1. L. Hong, A. K. Jain, and S. Pankanti, "Can multibiometrics improve performance?," in *Proceedings AutoID'99*, (Summit(NJ), USA), pp. 59–64, Oct 1999.
2. A. K. Jain, S. Prabhakar, and S. Chen, "Combining multiple matchers for a high security fingerprint verification system," *Pattern Recognition Letters*, vol. 20, pp. 1371–1379, 1999.
3. Y. Zuev and S. Ivanon, "The voting as a way to increase the decision reliability," in *Foundations of Information/Decision Fusion with Applications to Engineering Problems*, (Washington D.C., USA), pp. 206–210, Aug 1996.
4. R. Cappelli, D. Maio, and D. Maltoni, "Combining fingerprint classifiers," in *First International Workshop on Multiple Classifier Systems*, pp. 351–361, Jun 2000.
5. J. Kittler, M. Hatef, R. P. Duin, and J. G. Matas, "On combining classifiers," *IEEE Transactions on PAMI*, pp. 226–239, Mar 1998.
6. E. Bigun, J. Bigun, B. Duc, and S. Fischer, "Expert conciliation for multimodal person authentication systems using bayesian statistics," in *First International Conference on AVBPA*, (Crans-Montana, Switzerland), pp. 291–300, Mar 1997.
7. S. Ben-Yacoub, Y. Abdeljaoued, and E. Mayoraz, "Fusion of face and speech data for person identity verification," Research Paper IDIAP-RR 99-03, IDIAP, CP 592, 1920 Martigny, Switzerland, Jan 1999.
8. R. Brunelli and D. Falavigna, "Person identification using multiple cues," *IEEE Transactions on PAMI*, vol. 12, pp. 955–966, Oct 1995.
9. M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.
10. A. K. Jain, A. Ross, and S. Pankanti, "A prototype hand geometry-based verification system," in *Second International Conference on Audio and Video-based Biometric Person Authentication*, (Washington, D.C., USA), pp. 166–171, Mar 1999.
11. A. K. Jain, L. Hong, S. Pankanti, and R. Bolle, "An identity authentication system using fingerprints," in *Proceedings of the IEEE*, vol. 85, pp. 1365–1388, 1997.
12. A. K. Jain, A. Ross, and S. Prabhakar, "Fingerprint Matching Using Minutiae and Texture Features," to appear in the *Proceedings of ICIP 2001*, (Greece), Oct 2001.

PrimeEye: A Real-Time Face Detection and Recognition System Robust to Illumination Changes

Jongmoo Choi, Sanghoon Lee, Chilgee Lee, and Juneho Yi

School of Electrical and Computer Engineering
Sungkyunkwan University
300, ChunChundong, Jangan-gu, Suwon 440-746, Korea
{jmchoi, armofgod, cslee, jhyi}@ece.skku.ac.kr

Abstract. This research features a real-time face detection and recognition system named PrimeEye. The purpose of the system is for access control to a building or an office. The main feature of the system is face detection and face recognition robust to illumination changes. A simple adaptive thresholding technique for skin color segmentation is employed to achieve robust face detection. The system is also capable of operating in two different modes for face recognition: under normal illumination condition and under severe illumination changes. The experimental results show that the SKKUfaces method is better than the Fisherfaces method in the case of severe illumination changes. In the normal illumination condition, the Fisherfaces method is better than the SKKUfaces method.

1 Introduction

Security systems based on automatic face recognition have advantages over systems using other biometric identification techniques in that face recognition provides a more natural means of person identification. However, face detection and recognition in lighting variations is a very hard problem. We have developed a practical working system based on face detection and recognition for the purpose of application to access control to a building or an office.

The main features of the system are as follows. First, we have achieved face detection robust to illumination changes. We have developed a novel adaptive thresholding method for skin color segmentation by a simple characterization of the relationship between saturation and illumination values. Second, for face recognition, we have enhanced the recognition performance by enabling the system to not only work under normal lighting condition but also operate in a particularly tuned way when there are severe illumination changes.

In the following section, we give a brief overview of the entire system. Section 3 and 4 describe face detection and face recognition parts of the system respectively. We report the experimental results on the recognition performance of the system in section 5.

2 System Overview

The system can operate with a normal PC camera of 300,000 pixel resolution and the illumination condition for normal operation is interior lighting brighter than 60 lux. It takes less than 0.5 second to process an entire computation cycle from image capture to face recognition. The normal operating range from camera to a person to be identified is within 3m. If the distance from camera to a person is more than 3m, the system only tries to detect the person's face. The performance of the system is robust to illumination changes and eye glasses. Face rotation about 30 degrees in the image plane and partial occlusion of a face are also allowed. Fig.1 shows a block diagram of the entire system.

The system is divided into face detection and face recognition parts. The detection part consists of (1) skin color segmentation, (2) eye detection and (3) normalization of detected face regions based on the eyes location. The normalization refers to rotation, scaling and alignment of detected face regions to be input to the recognition part. For the recognition of detected faces, we have employed SKKUfaces method [6] that is an enhanced Fisherfaces method. The SKKUfaces method has two operating modes in order to tweaking the recognition performance: one is for normal illumination condition and the other for severe illumination changes.

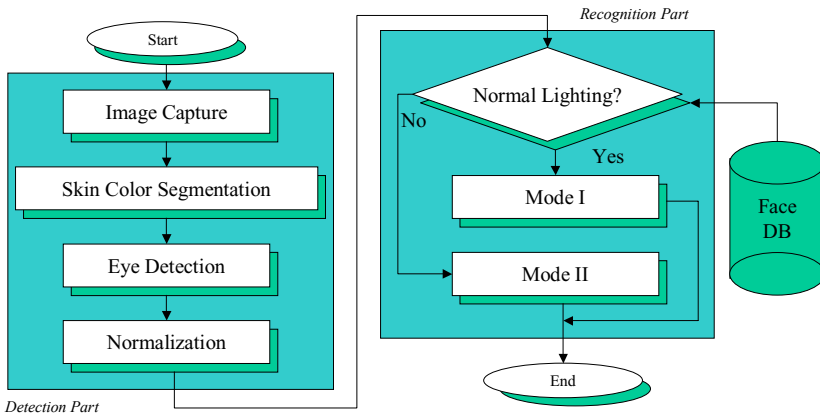


Fig. 1. System overview.

3 Face Detection

The first step is to select candidate face regions by skin color segmentation. After removing noise in the input image, we find connected components and compute their positions and sizes. We filter out non-facial regions by finding eyes. Finally, face regions are normalized for recognition using the eyes locations.

3.1 Skin Color Segmentation Robust to Illumination Changes

The HSI color model is commonly used in color image segmentation because it is relatively easy to separately consider the luminance and the chrominance components. However, it is well known that the luminance affects the saturation. Fig. 2 shows the cumulative distribution of HS values of ten faces from the same ethnic group. They form a relatively tight cluster in the HS color space that can be approximately modeled by an ellipse. As the illumination varies, the elliptical distribution also changes. We have observed the relation between luminance and saturation as shown in Fig. 3. Based on this observation, the system is implemented to adaptively set the threshold values for skin color segmentation when the luminance value changes.

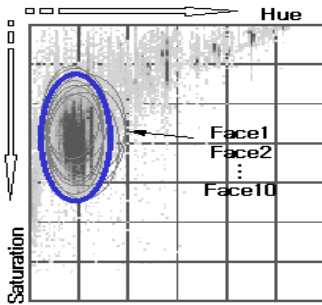


Fig. 2. Skin color distribution.

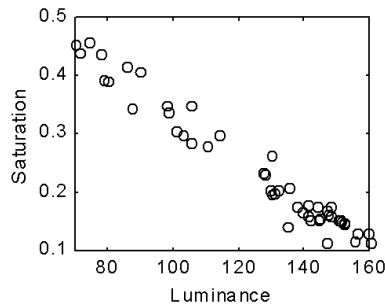


Fig. 3. Luminance vs. Saturation of skin color.

3.2 Detection of Face Regions

The regions obtained from skin color segmentation and connected component analysis may contain multiple faces or hands or other regions similar to skin color. We select only face regions using the position and size information of each region and some heuristics. The basic heuristic idea is that when there is no occlusion, the largest component tends to be the nearest face from the camera. In the case of partial occlusion, step edges between two faces are used to isolate them.

3.3 Eye Detection and Normalization of a Face Region

Eye detection is crucial in our system to judge a candidate face region as a real face region. As shown in Fig. 4 (a), a moving window searches for a black region, which corresponds to the pupil of the eye. The search location directs from down to top in the trajectory of an ellipse. We have learned from many trials that this way, the window locations do not go into the regions of hair or eyebrows when the face is rotated. As can be seen in Fig. 4 (b) - (d), eyes are correctly detected and located. The normalization step is composed of linear transformation, histogram equalization and

masking. Using the location of detected eyes, the linear transformation rotates and scales the face to the size of 50 x 50 image. The histogram equalization is employed to make the facial image have good contrast. Lastly, the hair style and the background are removed as much as possible to only obtain the pure facial part. We call this masking process.

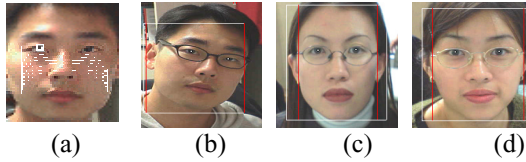


Fig. 4. Eye detection.

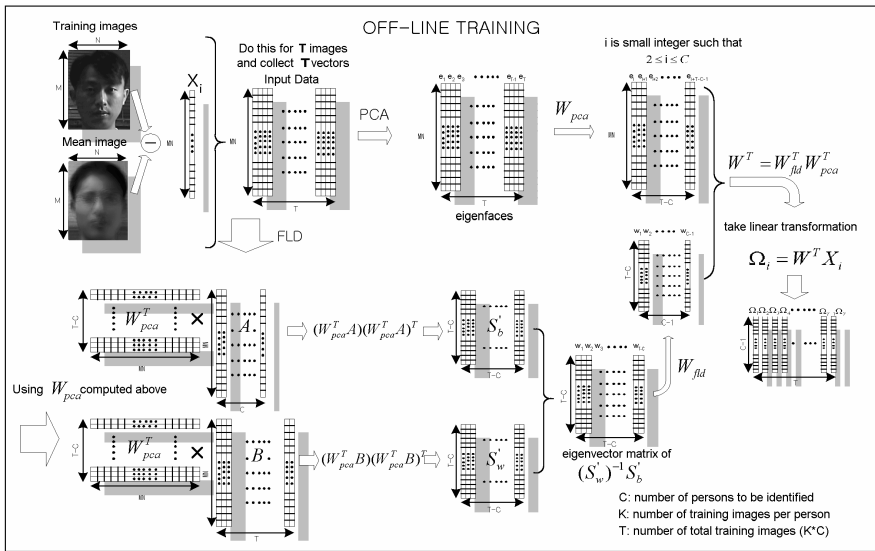


Fig. 5. The training phases of the Fisherfaces ($i=1$) and the SKKUfaces ($i \geq 2$).

4 Face Recognition

For the recognition of faces, we have employed the SKKUfaces method [6] illustrated in Fig. 5. The SKKUfaces method is similar to the Fisherfaces method in that it first performs PCA (principal component analysis) and apply LDA (linear discriminant analysis) to the eigen-representation computed by PCA. It is well known that the PCA step is a requirement in Fisherfaces-like approaches to guarantee that between-class

scatter matrix be invertible. The SKKUfaces method is different from the Fisherfaces method in that it applies LDA to the eigen-representation computed by PCA with first few eigenvectors dropped. The effect is that face variations due to severe illumination changes are effectively removed but important information for the recognition of faces under normal illumination condition may be lost as well. Another advantage of SKKUfaces is the efficient computation of the between-class scatter and within-class scatter matrices [6].

Our system works in two different modes. One is the Fisherfaces mode that is supposed to be used for the recognition of faces under normal illumination condition. The other is the SKKUfaces mode for face recognition under severe illumination changes. In the SKKUfaces mode, only the top eigenvector is eliminated in the current implementation of the system. The system detects normal lighting condition basically by checking the symmetry of brightness pattern of a facial image.

5 Experimental Results

We have compared the recognition performance of Eigenfaces [3], Fisherfaces and SKKUfaces methods by using two different kinds of facial image database.

5.1 Experiment I: Recognition under Severe Illumination Changes

We use two image sets, SKKU database [7] and YALE database [8]. The size of a facial image is 50×40 . The SKKU database consists of facial images captured in simple backgrounds, which are gathered under variations of luminance, facial expression, glasses, and time interval. The database contains 100 images of 10 persons. The YALE database is constructed in a similar fashion as the SKKU database. As shown in Table 1, the SKKUfaces method has lower error rate. We can see that the SKKUfaces method performs better than the others in the case of severe illumination changes. We have also found that the SKKUfaces method has computational advantages of reducing space and time complexity over the Fisherfaces method. The SKKUfaces method only need to compute matrices of size of 2000×100 and 2000×10 for within-class covariance and between-class covariance, respectively, while the other methods require computation of matrices of size 2000×2000 .

5.2 Experiment II: Recognition under Normal Illumination Changes

Facial images of size 50×50 are obtained from eight people under lighting conditions (60~380lux), facial expression, pose (swing within 30° degree). The total number of images is 230 images composed of 15~40 images per person. We have used 115 images for training and the others for testing. We have examined FRR (false reject Rate) and FAR (false accept rate). The error rates used for comparison is EER (equal error rate) that is the point where the values of FRR and FAR are equal. As shown in Table 1, the Fisherfaces and the SKKUfaces methods are better than the Eigenfaces method and the Fisherfaces method has lower ERR than the SKKUfaces method.

5.3 Discussion

The experimental results show that the SKKUfaces method is more appropriate to use than the Fisherfaces method in the case of severe illumination changes. We can see that face variations due to severe illumination changes are effectively removed sacrificing some information for the recognition of faces. In the normal illumination condition, the Fisherfaces method is the one that has to be used for face recognition.

Table 1. Error rates for the three methods (%).

Method	Experiment I (FRR)		Experiment II (EER)
	SKKU DB	YALE DB	
Eigenfaces	45	30	9.4
Fisherfaces	12	8	1.1
SKKUfaces	9	4	2.0

6 Conclusions

We have presented a real-time system for face detection and recognition for the application to access control to a building or an office. The system has been designed and implemented focusing on robustness to lighting variations and processing speed. The performance of the system will be enhanced with the development of a more intelligent method of switching the recognition modes according to the lighting condition.

References

1. R.S. Berns, Principles of Color Technology, John Wiley & Sons Inc., pp. 22-23, 2000.
2. P. Belhumeur, J. Hespanha, and D. Kriegman, Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, IEEE Trans. on PAMI, vol.19,no. 7, pp. 711- 720, 1997.
3. M. Turk and A. Pentland, Eigenfaces for Recognition, Journal of Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.
4. M. Kirby and L. Sirovich, Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces, IEEE Trans. on PAMI, vol .12, no .1, pp. 103-108, 1990.
5. K. Etemad and R. Chellappa, Discriminant Analysis for Recognition of Human faces image, Journal of Optical Society of America, vol. 14, no. 8, pp. 1724-1733, 1997.
6. J. Yi, H. Yang and Y. Kim, Enhanced Fisherfaces for Robust Face Recognition, Lecture Notes in Computer Science, Vol. 1811, pp. 50 2-511, 2000.
7. <http://vulcan.skku.ac.kr/research/skkufaces.html>.
8. <http://cvc.yale.edu/projects/yalefaces/yalefaces.html>.

A Fast Anchor Person Searching Scheme in News Sequences

Alberto Albiol¹, Luis Torres², and Edward J. Delp³

¹ Politechnic University of Valencia

Crta Nazaret-Oliva S/N, 46730 Grao de Ganda, Spain

alalbiol@com.upv.es

<http://ttt.gan.upv.es/~alalbiol>

² Politechnic University of Catalonia, Barcelona, Spain

³ Purdue University, West Lafayette, USA

Abstract. In this paper we address the problem of seeking anchor person shots in news sequences. This can be useful since usually this kind of scenes contain important and reusable information such as interviews. The proposed technique is based on our a priori knowledge of the editing techniques used in news sequences.

1 Introduction

The amount of digital video has undergone an explosive growth in the last years. One of the problems when dealing with digital video is the huge amount of data to be analyzed. This problem can be minimized if we are able to select relevant portions of the video where more powerful analysis tools can be applied.

In broadcast news, around 85% of the time consists of a television announcer or an off-voice of a journalist reading a report, while only the remaining 15% consists of anchor person shots of interviewed people. We are interested in locate these shots where more sophisticated tools such as speaker or face recognition can be applied. However, in order to efficiently apply these techniques a tool for seeking interviewed people is needed. In this paper we address this problem. Our approach takes advantage of our a priori knowledge of the editing techniques used in news sequences. In section 2 the main elements that build a news sequence are described, this will give us the key idea to the presented approach. Sections 3 to 5 present the analysis tools that will be used. Some results are provided in section 6.

2 Elements of a News Sequence

News sequences are composed usually of the following elements:

1. Headings.
2. Live journalist speeches. Usually the images are either a close-up of the television announcer or topic related scenes.

3. Prerecorded videos, usually containing the journalist off-voice while some related images are displayed. Also, short anchor person scenes of interviewed people are usually inserted in the videos.

Figure 1 illustrates the previous scenes types. In this paper we are interested in locating interviewed people as in 1.d. The editing procedure for this type of scenes can be summarized as follows:

1. The reporter records his/her voice but no image is recorded yet.
2. If an interview is going to be inserted, then its audio and video are inserted together, creating a simultaneous audio and video cut.
3. If the reporter needs to continue with his/her speech or more interviews need to be inserted, the two first steps are repeated.
4. Finally, images are added for the reporter voice periods, usually several shots are inserted for each audio segment.

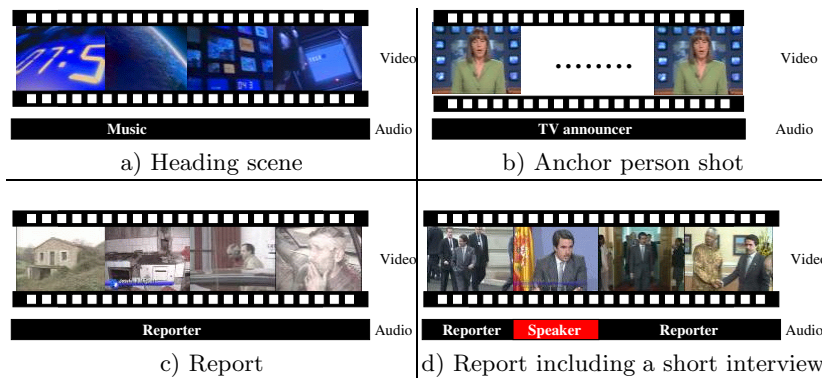


Fig. 1. News sequence elements.

The consequence of this editing procedure is that interviews can be easily detected studying the matching of audio and video cuts. Sections 3 and 4 will describe the algorithms to detect audio and video cuts, while in section 5 a procedure to combine those results will be proposed.

3 Audio Segmentation

The goal of speaker segmentation is to locate all the boundaries between speakers in the audio signal. Some speaker segmentation systems are based on silence detection [1]. These systems rely on the assumption that utterances of different people are separated by significant silences. However reliable systems would require cooperative speakers which is not the case for broadcast news. Other segmentation approaches are based on speaker turn detection. These systems aim

to segment the audio data into homogeneous segments containing one speaker only. Usually a two-step procedure is used, where the audio data is first segmented in an attempt to locate acoustic changes. Most of these acoustic changes will correspond to speaker turns. The second step is used then to validate or discard these possible turns. The segmentation procedures can be classified into three different groups: phone decoding [2, 3], distance-based segmentation [4, 5], hypothesis testing [6]. In this paper we will use the DISTBIC algorithm [5] to partition the audio data. DISTBIC is also a two-step segmentation technique. In the first step distance between adjacent windows is obtained every 100ms. This result in a distance signal $d(t)$, see Figure 2. In our implementation we

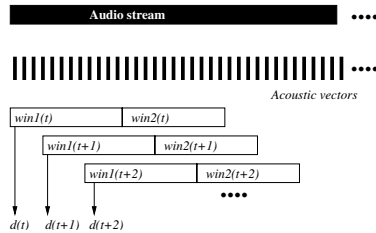


Fig. 2. Sliding windows.

use the symmetrical kullback-Leibler [4] distance. The significant peaks of $d(t)$ are considered as turn candidates. In the second step the turn candidates are validated using the BIC criteria [7]. To that end, the acoustic vectors of adjacent segments are modeled separately using Gaussian models. The model of the union of the acoustic vectors of both segments is also computed and then the BIC criteria is used to check if the likelihood of the union is greater than the likelihood of both segments individually. In the case that the likelihood of the union is greater then the turn point is discarded. Otherwise the turn point is validated.


4 Video Segmentation

Several techniques have been proposed in the literature for detection of cuts in video sequences [8, 9, 10]. Most of them rely on the similarity of consecutive frames. A popular similarity measurement is the *mean absolute frame difference (MAFD)*:

$$MAFD(n) = \frac{1}{IJ} \sum_{i=1}^I \sum_{j=1}^J \clubsuit_n(i^c j) \otimes f_{n \otimes 1}(i^c j) \clubsuit \tag{1}$$

where I and J are the horizontal and vertical dimensions of the frames, n is the frame index, and $(i^c j)$ are the spatial coordinates.

Other popular similarity measures include the displaced frame difference (DFD), which reduces the contribution of camera and objects motion, at the

expense of a greater computational load. In this work, techniques that need motion estimation are avoided because of the higher computational requirements. Also low resolution images obtained from the DC coefficients of the MPEG compressed video stream will be used to compute the MAFD measurement. This has the advantage that does not require full decompression of the video in order to find the cuts .

The causes of dissimilarity between consecutive frames include:




- Actual scene transitions.
- Motion of objects.
- Camera motion.
- Luminance variations (flicker).

In standard (good condition), the last contribution is normally negligible (except for special situations such as the presence of flashlights). Motion of objects and camera normally occur during more than one transition which produces wide pulses in the MAFD signal. On the other hand, an abrupt scene transition produces a peak of width one in MAFD. This difference can be exploited to distinguish motion and cuts in video. Basic morphological operations, such as openings and closings, can be applied for this purpose. The proposed algorithm for cut detection can be summarized as follows:

- Obtain $\text{MAFD}(n)$.
- Compute the residue of the morphological opening of $\text{MAFD}(n)$.
- Threshold the residue to locate the cuts.

We are well aware that more sophisticated video cut detection algorithms exist. However, this simple algorithm provides very good results, since other transitions effects such as wipes or fades are not usually used in news reports. Moreover, the interviews do not usually show a high shot activity (usually the scene is an anchor person), therefore the false alarm rate within these intervals is nearly zero.

5 Audio and Video Correspondence

Once the audio and video segments are located the objective is to find the correspondence between them. Figure a shows the ideal situation that we are trying to find, i.e. the audio and video segments overlap. However, for real sequences the borders of audio and video segments do not overlap, as shown in figure b. This is due mainly because silence periods are usually located in the audio segment borders creating a small inaccuracy. Figure c shows an example of the typical situation for report segments, where a long audio segment coexists with short video segments. Given an audio segment in the time interval $[t_{min1}, t_{max1}]$ and a video segment defined in the interval $[t_{min2}, t_{max2}]$. The intersection interval is defined as:

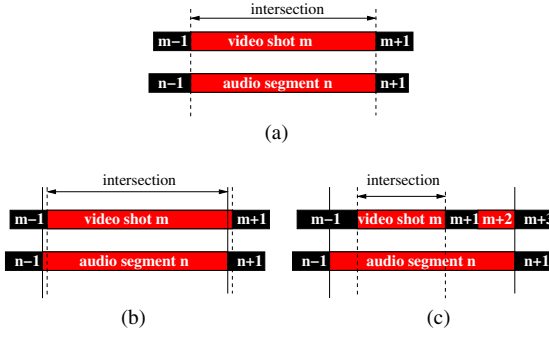


Fig. 3. (a) Audio and video borders math exactly. (b) Audio and video borders almost match. (c) Audio segment contains several video shots.

$$[t_{min} \cap t_{max}] = [\max(t_{min1} \cap t_{min2}) \cap \min(t_{max1} \cap t_{max2})] \quad (2)$$

then if $(t_{max} \cap t_{min}) > 0$ for a pair of audio and video segments, we define the overlap degree as:

$$overlap = \min \left[\frac{(t_{max} \cap t_{min})}{(t_{max1} \cap t_{min1})}, \frac{(t_{max} \cap t_{min})}{(t_{max2} \cap t_{min2})} \right] \quad (3)$$

If $overlap > 0.9$ then the audio and video segments are said to match and a new index entry is created.

6 Results and Conclusions

The previous algorithms have been tested on several 30 minutes news sequences. The results have been evaluated with the following parameters: Detection Rate (DR):

$$DR = 100 \cdot \frac{\text{number of detected interviews}}{\text{number of actual interviews}} \quad (4)$$

False alarm rate (FAR):

$$FAR = 100 \cdot \frac{\text{number of false alarms}}{\text{number of actual interviews} + \text{number of false alarms}} \quad (5)$$

and Selected Time (ST)

$$ST = \frac{\text{total duration of the selected shots}}{\text{Sequence duration}} \quad (6)$$

Table 1 presents some results. It can be seen how the algorithm allows to discard a large portion of the sequence from consideration with minimal processing. Almost all false detected shots correspond to anchor person shots where the speaker is a reporter.

Table 1. Results.

DR	FAR	ST
94 %	41 %	31 %

We have presented a novel fast algorithm to detect interviews without needing to analyze in detail the whole sequence. Once the segments of interest are located more sophisticated analysis tools can be used such as: speaker or face recognition. These analysis tools can be used independently or they can also be combined to obtain more reliable results.

References

1. M. Nishida and Y. Ariki, "Speaker indexing for news, articles, debates and drama in broadcasted tv programs," in *IEEE International Conference on Multimedia, Computing and Systems*, 1999, pp. 466–471.
2. T. Hain, S. E. Johnson, A. Tuerk, P. C. Woodland, and S. J. Young, "Segment generation and clustering in the htk broadcast news transcription system," in *Proceedings of DARPA Broadcast News Transcription Understanding Workshop*, Landsdowne, VA, 1998, pp. 133–137.
3. D. Liu and F. Kubala, "Fast speaker change detection for broadcast news transcription and indexing," in *Proceedings ESCA Eurospeech'99*, Budapest, Hungary, 1999, vol. 3, pp. 1031–1034.
4. M. A. Siegler, U. Jain, B. Raj, and R. M. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *Proceedings of the DARPA speech recognition workshop*, Chantilly, Virginia, February 1997, pp. 97–99.
5. P. Delacourt and C. J. Wellekens, "Distbic: A speaker-based segmentation for audio indexing," *Speech communication*, vol. 32, no. 1-2, pp. 111–127, September 2000.
6. S. Wegmann, P. Zhan, and L. Gillick, "Progress in broadcast news transcription at dragon systems," in *Proceedings of International Conference on Acoustics Speech and Signal Processing*, Phoenix, AZ, 1999, pp. 33–36.
7. S. S. Chen and P. S. Gopalakrishnan, "Speaker environment and channel change detection and clustering via de bayesian information criterion," in *DARPA Speech Recognition Workshop*, 1998.
8. J. S. Boreczky and L. A. Rowe, "Comparison of video shot boundary detection techniques," in *Proceedings SPIE Conference on Visual Communications and Image Processing*, 1996.
9. A. Albiol, V. Naranjo, and J. Angulo, "Low complexity cut detection in the presence of flicker," in *Proceedings 2000 International Conference on Image Processing*, Vancouver, Canada, October 2000.
10. B. Liu and B. Yeo, "Rapid scene analysis on compressed video," *IEEE Transactions on Circuits and Systems*, vol. 5, no. 6, pp. 533–544, September 1995.

Author Index

- Albiol A. 366
- Bartesaghi A. 259
- Bazen A.M. 198
- BenAbdelkader C. 284
- Bhanu B. 205
- Bigun J. 44, 247
- Bobick A.F. 301
- Bolle R.M. 223
- Brand J.D. 157
- Büke B. 336
- Byun H. 32
- Carter J.N. 272, 278
- Castrillón M. 59
- Chindaro C. 84
- Choi J. 360
- Choy K.-W. 44
- Chua C.S. 26
- Chung Y. 32
- Clark A.F. 14
- Cocquerez J.P. 121
- Colomb S. 157
- Connell J.H. 223
- Cruz-Llanas S. 217
- Cutler R. 284
- Damper R.I. 169
- Davis J.W. 295
- Davis L. 284
- Davoine F. 121
- Delp E.J. 366
- Déniz O. 59
- Deravi F. 84
- Dubuisson S. 121
- Erçil A. 336
- Farup I. 65
- Feitosa R.Q. 71
- Fernández A. 259
- Foster J.P. 312
- Fränti P. 150
- Frasconi P. 241, 253
- Frischholz R. W. 90
- Fröba B. 78
- Garca Mateos G. 102
- Garg G. 192
- Gerez S.H. 198
- Ghaderi R. 1
- Gillies D.F. 71
- Gómez A. 259
- Glez-Rodriguez J. 217
- Gowdy J.N. 175
- Guo Y. 52, 96
- Gurbuz S. 175
- Gutta S. 38
- Hamamoto T. 108
- Hangai S. 108
- Hayfron-Acquah J.B. 272
- Hernández M. 59
- Higgins J.E. 169
- Hjelmås E. 65
- Ho Y.-K. 26
- Huang J. 38
- Jain A.K. 182, 211, 354
- Jesorsky O. 90
- Johnson A.Y. 301
- Kim J. 235
- Kim S. 235
- Kinnunen T. 150
- Kirchberg K.J. 90
- Kırmızıtaş H. 336
- Kittler J. 1
- Komiya Y. 318
- Koo W.M. 229, 266
- Korczak J. 348
- Kot A. 229, 266
- Küblbeck C. 78
- Lee C. 360
- Lee D. 235
- Lee K. 32
- Lee S. 360
- Lincoln M.C. 14
- Liu C. 20, 38
- Marcialis G.L. 241

- Mariani R. 115
Mason J.S.D. 157
Matas J. 1
Matsumoto T. 318
Morita H. 318

Nakamura S. 127
Nanda H. 284
Nilsson K. 247
Nixon M.S. 272, 278, 312

Öden C. 336
Ohishi T. 318
Olsson H. 44
Omata M. 108
Ortega-Garcia J. 217

Pankanti S. 182, 211
Patterson E. 175
Pelecanos J. 144
Poh N. 348
Pontil M. 253
Prabhakar S. 182, 211
Prugel-Bennett A. 312

Qian J.-Z. 354

Rajapakse M. 96
Ratha N.K. 223
Roli F. 241

Ross A. 182, 354

Sakamoto T. 318
Sanchez-Avila C. 324, 330
Sanchez-Bote J.L. 217
Sanchez-Reillo R. 324, 330, 342
Sharma P. K. 192
Shen L.L. 266
Simon-Zorita D. 217
Sullivan K.P.H. 144

Tan X. 205
Thomaz C.E. 71
Torres L. 366
Tufekci Z. 175

Udupa U.R. 192

Vicente Chicote C. 102

Wang Y. 26
Wechsler H. 20, 38
Windeatt T. 1

Yam C.-Y. 278
Yao Y. 253
Yi J. 360
Yıldız V.T. 336

Zhang B.L. 52